Auction-Based Efficient Online Incentive Mechanism Designs in Wireless Networks

Gang Li

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy (Electrical and Computer Engineering) at

Concordia University

Montréal, Québec, Canada

October 2021

CONCORDIA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Mr. Gang Li

Entitled: Auction-Based Efficient Online Incentive Mechanism Designs in Wireless Networks

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Electrical and Computer Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

______________________________ Chair
Dr. Kudret Demirli

______________________________ External Examiner
Dr. Lian Zhao

______________________________ External to Program
Dr. Wen-Fang Xie

______________________________ Examiner
Dr. Mustafa K. Mehmet Ali

______________________________ Examiner
Dr. Dongyu Qiu

______________________________ Supervisor
Dr. Jun Cai

Approved by ______________________________
Dr. Wei-Ping Zhu, Graduate Program Director

October 29, 2021

____
Dr. Mourad Debbabi, Dean
Gina Cody School of Engineering & Computer Science

Abstract

Auction-Based Efficient Online Incentive Mechanism Designs in Wireless Networks

Gang Li, Ph.D.

Concordia University, 2021

Recently, wide use of mobile devices and applications, such as YouTube and Twitter, has facilitated every aspect of our daily lives. Meanwhile, it has also posed great challenges to enable resource-demanding users to successfully access networks. Thus, in order to enlarge network capacity and fully make use of vacant resources, new communication architectures emerge, such as D2D communications, edge computing, and crowdsourcing, all of which ask for involvement of end mobile users in assisting transmission, computation, or network management. However, end mobile users are not always willing to actively provide such sharing services if no reimbursements are provided as they need to consume their own computation and communication resources. Besides, since mobile users are not always stationary, they can opt-in and opt-out the network for their own convenience. Thus, an important practical characteristic of wireless networks, i.e., the mobility of mobile users cannot be ignored, which means that the demands of mobile users span over a period of time. As one of promising solutions, the online incentive mechanism design has been introduced in wireless networks in order to motivate the participation of more mobile users under a dynamic environment. In this thesis, with the analyses of each stakeholder's economic payoffs in wireless networks, the auction-based online incentive mechanisms are proposed to achieve resource allocations, participant selections, and payment determinations in two wireless networks, i.e., Crowdsensing and mobile edge computing. In particular, i) an online incentive mechanism is designed to guarantee Quality of Information of each arriving task in mobile crowdsensing networks, followed by an enhanced online strategy which could further improves the competitive ratio; ii) an online incentive mechanism jointly considering communication and computation resource allocations in collaborative edge computing networks is proposed based on the primal-dual theory; iii) to deal

with the nonlinear issue in edge computing networks, an nonlinear online incentive mechanism under energy budget constraints of mobile users is designed based on the Maximal-in-Distributional Range framework; and iv) inspired by the recent development of deep learning techniques, a deep incentive mechanism with the budget balance of each mobile user is proposed to maximize the net revenue of service providers by leveraging the multi-task machine learning model. Both theoretical analyses and numerical results demonstrate the effectiveness of the designed mechanisms.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor Prof. Jun Cai for his continuous support during my Ph.D. study. Prof. Cai is always leaving me the time to discuss many questions with him and he helped me revise papers very patiently and carefully. Thank you for your mentorship and support. I hope we can continue to collaborate in the future. Besides, I would like to thank my examining committee members: Prof. Dongyu Qiu, Prof. Wen-Fang Xie, and Prof. Mustafa K. Mehmet Ali for their efforts and time. Their feedbacks and constructive suggestions have been invaluable for the improvement of my work. A special thank is given to Prof. Lian Zhao (Ryerson University, Toronto, ON, Canada) for her time to read my thesis. In addition, I would also like to express my sincere thanks to Prof. Hongbin Chen for his encouragement and invaluable support. It is Prof. Hongbin Chen who leads me to the road of scientific research. All of these are very helpful in finishing my research work and the writing of this thesis.

I thank all my colleagues in the Network Intelligence and Innovation Lab (NI2L) for their insightful comments and suggestions during our discussions. My sincere thanks go also to the teaching, administrative and technical staff of the department of electrical and computer engineering, the Concordia University, which provides a peaceful and productive working environment.

Last but not least, I must express my profound gratitude to my wife, Min Zhou, for her continuous support and genuine dedications, and to my parents, who are always behind me and offering me with their selfless supports. Without them, I couldn't go so far and this thesis would not have been completed. I also appreciate the continued encouragement that my dear friends gave me. I acknowledge the University of Manitoba and Concordia University for awarding me the International Graduate Student Scholarship and the Tuition Award of Excellence, respectively.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviations	Full Name
CE	Computational Efficiency
CR	Competitive Ratio
IC	Incentive Compatibility
IR	Individual Rationality
SW	Social Welfare
BB	Budget Balance
QoI	Quality of Information
VM	Virtual Machine
ETSI	European Telecommunications Standards Institute
AI	Artificial intelligence
BS	Base Station
EC	Edge Computing
D2D	Device-to-device
CPU	Central Processing Unit
KKT	Karush-Kuhn-Tucker
CSCG	Circularly Symmetric Complex Gaussian
TSW	Total Social Welfare
FIFO	First In First Out
SNR	Signal-to-Noise Ratio
VCG	Vickrey-Clarke-Groves

MDP	Markov Decision Process
LOS	Lehmann-Ocallaghan-Shoham
IoT	Internet of Things
MIDR	Maximal-in-Distributional Range
LPWAN	Low-power Wide Area Networking
Lyapunov	Lyapunov
OTM	One-shot Truthful Mechanism
IRSM	Integrate Rounding Scheme based MIDR
BIC	Bayesian Incentive Compatibility
DSIC	Dominate-Strategy Incentive Compatibility

Chapter 1

Introduction

In past decades, many researches have put their focuses on proposing novel system models and designing efficient resource allocation algorithms in wireless networks. However, in beyond 5G networks, it is anticipated that 80 billion devices will be connected to the Internet by 2025, and the global data will reach 163 zettabytes, which is 10 times the data generated in 2016 [1]. This forces network designers to enlarge network capacity by reusing and sharing mobile users' dedicated resources. However, mobile users may not be willing to share and reuse their resources with others as they will consume energies, computation, and communication resources. Moreover, in many practical scenarios, the considered networks usually operate within a dynamic setting, which means decision-makings should be completed on-the-fly. As a consequence, the study of designing online incentive mechanisms in wireless networks become mandatory and promising. In this thesis, we focus our studies on designing online incentive mechanisms to address resource allocation issues in two wireless networks (crowdsensing networks and edge computing networks), in which four different topics are discussed, i.e., random task arrival in crowdsensing, online mechanism design in collaborative edge computing, nonlinear online mechanism design for task offloading edge computing, and machine learning based mechanism design in edge computing. In the following subsections, we will introduce the background and some basics of incentive mechanism designs, and provide our research motivations.

1.1 Backgrounds in Incentive Mechanism Designs

In this section, the development of incentive mechanisms is first described. Then, some important concepts and terminologies in incentive mechanism designs will be introduced.

1.1.1 Development of Incentive Mechanism Designs

Incentive mechanism design stipulates a set of rules for the participators to behave according to the designer's goal. According to [2], incentive mechanism designs can be roughly classified into four categories, which are auctions, contracts, lotteries, and trust and reputation systems. It is generally believed that incentive mechanism design can be originated to 1961 when the paper was published by Vickrey William [3]. Since then, the first theoretical research surge on incentive mechanism designs was raised in the mid 1980s. The initial marriage of mechanism design with computer science was in around 1980s [4], and later developed a new theory, called algorithmic mechanism design, in computer science. In telecommunication systems, the pioneer work by using auction was in earlier 1990s, when the paper [5] was published to address the access and handoff issue in the wireless personal communication system. The concept of "online" means that the decisions are made based on the current and past information and without any future knowledge. Online algorithms have been implicitly studied for almost half a century in the fields of optimizations, data structures, scheduling, routing in communication networks, and other areas. The traditional approach to studying online algorithms falls into the framework of distribution complexity, whereby someone assumes a distribution on certain input information, and studies the expected total payoff for each event. Since about 30 years ago, the focus on this method has been renewed and shifted to the approach called competitive analysis [6], in which the performance of the online algorithm is measured by comparing its performance to that of optimal offline counterpart.

Online incentive mechanisms extend the traditional offline incentive mechanism to be applicable in a dynamic environment. In the online incentive mechanism, decisions on allocations and payments are made as soon as the private information of bidders is revealed without knowing the future information *a prior*. Even though a lot of events about online auction are happened or held in the past few years, we only summarize some important and meaningful activities or research

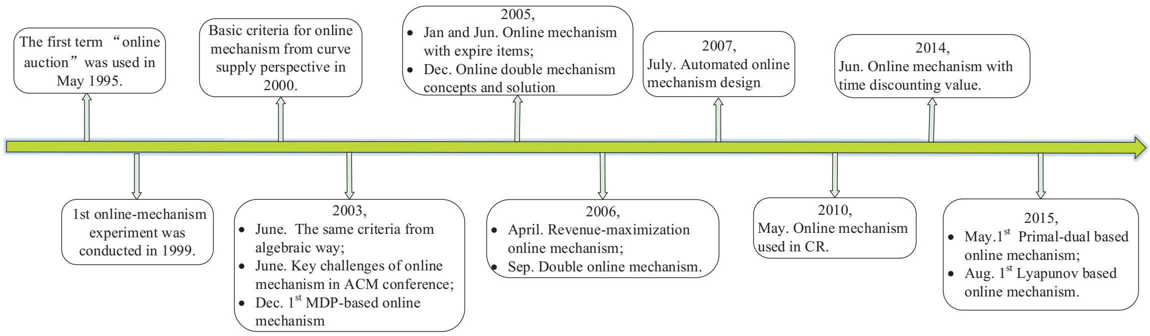

Figure 1.1: Chronological development of online incentive mechanisms.

results, which are drawn in Fig. 1.1. The term “online auction” was coined by the company Onsale in May, 1995, which was designed to sell commercial goods over the internet. Four years later, the first experiment on the collectible trading cards was auctioned off over the internet. The earliest paper [7] about modern online auction was published in 2000 where the basic requirements for designing an online truthful auction was derived by the study of the supply-demand curve. In 2003, these basic requirements were further corroborated in [8] from the algebraic perspective. Almost at the same time, the key challenges by designing an online auction were pointed out in [9]. In December of this year, the first online auction based on Markov Decision Process (MDP) was proposed by Parkes in [10]. In 2005, by generalizing the results of Mcafee, a preliminary solution on double online auction to maximize the social welfare was proposed in [11]. In comparison, one year later, a more comprehensive solution with a theoretic performance guarantee was put forward for double online auction in [12]. Note that all the above work concentrated on designing online truthful auction to maximize the social welfare, while in 2006, Pai and Vohra [13] advanced the study of revenue-optimal online mechanisms in model-based environments, and together with work [14] to extend Myerson’s optimal auction to dynamic environments. Currently, the most popular designing methodology for online auction are primal-dual based method [15] and Lyapunov based method [16], which were first appeared on May and August of 2015, respectively.

1.1.2 Basics In Incentive Mechanism Designs

This section provides basic concepts and properties in designed mechanisms, and basic terminologies in auctions. Mechanism design is a subfield of game theory, which is used to define the game rules so as to achieve the desired outcome. Formally, a mechanism should consists of the following ingredients:

- Players $i \in \mathcal{N}$ with cardinality of $|\mathcal{N}| = N$ and preference types $b_i \in \mathcal{B}_i$;
- A strategy space $\mathcal{S} = \mathcal{S}_1 \times \cdots \times \mathcal{S}_N$, where player i chooses strategy $s_i(b_i) \in \mathcal{S}_i$;
- Utility $u_i(s_i(b_i), \boldsymbol{s}_{-i}(\boldsymbol{b}_{-i}))$, where $\boldsymbol{s}_{-i}(\boldsymbol{b}_{-i})$ is the set of strategies excluding player i.

In follows, for notation convenience, we directly use $u_i(b_i, \boldsymbol{b}_{-i})$ to denote $u_i(s_i(b_i), \boldsymbol{s}_{-i}(\boldsymbol{b}_{-i}))$. Traditionally, the following two important properties must be satisfied in the mechanism design.

Definition 1 (**Incentive Compatibility (IC)**)**.** *An incentive mechanism is incentive-compatible if for any type $b_i = v_i$, it is the dominant strategy regardless of other players' type, where v_i is the true type of player i. Then the incentive compatibility requires*

$$u_i(v_i, \boldsymbol{b}_{-i}) \geq u_i(b_i, \boldsymbol{b}_{-i}). \tag{1}$$

In real-world scenarios, this property is of great importance as it can force each player to play with their true type. Note that incentive compatibility is also known as truthfulness.

Definition 2 (**Individual Rationality (IR)**)**.** *An incentive mechanism is individually rational if the utility of each player is non-negative, i.e.,*

$$u_i(\boldsymbol{b}) \geq 0. \tag{2}$$

The property of individual rationality can guarantee that each player can obtain non-negative benefit if they are willing to participate.

Auctions are known as mechanisms, which are the most primary leverages to design incentive mechanisms in wireless networks. Therefore, we refer to the mechanism design as the auction design, and these two concepts are interchangeably used in this thesis. The history of auction theory in wireless networks dates from the application of spectrum license distribution. In general, auctions consist of two major components, which are the allocation rule, i.e., $\mathcal{S}$, and the payment rule (pricing policy), i.e., $\mathcal{P}$. Moreover, in most auctions, players are required to submit their types, and the winners and corresponding rewards will be made based on the designed allocation and payment rules. In follows, some basic terminologies and definitions are introduced in auctions.

- Auctioneer: An auctioneer is as an intermediate agent that implements allocation and payment rules in the auction. In wireless networks, auctioneer could often be the Base Station (BS);
- Seller: A seller offers its items for sale. The items could be spectrum, bandwidth, and computation resources in wireless networks;
- Bidder: A bidder is a buyer that wants to buy the items from the seller.
- Bid: The bid typically consists of the price that the bidder is willing to pay and its corresponding demanding items.

Definition 3 (Revenue). *The revenue is defined as the sum of the payments received from sellers, i.e.,*

$$\sum_{i \in \mathcal{N}} p_i x_i, \tag{3}$$

where p_i is the payment by bidder i; x_i is a binary variable, which means if bidder i wins, $x_i = 1$; otherwise, $x_i = 0$.

In incentive mechanism designs, the revenue maximization is usually applied as an objective to measure the benefits that sellers can earn. To further assess the benefits of the whole considered system, the Social Welfare (SW) maximization is commonly-used as another objective.

Definition 4 (SW). *The social welfare is defined as the summation of utilities in the auction including all bidders and sellers. By cancelling payments, the social welfare in the auction can be expressed as*

$$\sum_{i \in \mathcal{N}} v_i x_i. \tag{4}$$

Note that SW maximization is the most commonly considered objective in auctions.

Definition 5 (**Budget Balance (BB)**). *An incentive mechanism is budget balanced if the payments made by any bidder do not exceed their own budgets.*

In order to evaluate the performance of designed online incentive mechanisms, the metric Competitive Ratio (CR), defined as follows, is used to denote the worst performance that designed algorithms can achieve. Note that the competitive analysis is the worst case measurement so that the theoretical CR of an online algorithm may be much worse than the corresponding ratio observed in practice.

Definition 6 (**CR**). *An online algorithm $\mathcal{A}$ to a minimization problem (or maximization problem) is α-competitive if $\mathcal{A}(I) \leq \alpha OPT_{offline}(I)$ (or $\mathcal{A}(I) \geq \alpha OPT_{offline}(I)$), where $\alpha \geq 1$ is a constant, called CR, I is a sequence of input, and $\mathcal{A}(I)$ is the objective value by online algorithm $\mathcal{A}$ while $OPT_{offline}(I)$ is the optimum by an optimal offline algorithm.*

Definition 7 (**Online Incentive Mechanism (OIM)**). *An online incentive mechanism, i.e., $M = (\mathcal{S}, \mathcal{P})$, should solve $\mathcal{S}$ and $\mathcal{P}$ along time, and the designed online incentive mechanism should satisfy the requirements of IC, IR, and has a theoretic CR.*

In the following chapters, these aforementioned definitions will be applied, and some of which will be expressed in a concrete mathematical form based on various applications.

1.2 Motivation and Objectives

In the literature, previous efforts have put their focuses on proposing novel system models and designing resource allocation algorithms in wireless networks for decades. The recent upsurge of mobile devices, such as mobile users and tablets, has caused scarcity of network resources, such as spectrum and computation resources, which prevents mobile devices to successfully access networks. To overcome this dilemma, the concepts of sharing and reusing resources are putting forwarded to enlarge the capacity of wireless networks. For example, in D2D networks, with careful and efficient resource allocation and interference managements, some mobile users in a heavy-loaded wireless networks can still access networks by using licensed bandwidths, which are dedicated to other mobile users. However, these dedicated mobile users may not always be willing to participate in such assistance if no rewards are provided, because they inevitably suffer from costs in terms of energy consumption and communication/computation resource usage. Moreover, since mobile users are commonly intelligent and selfish, they may try to maximize their own benefits strategically, which should be avoided from the network management point of view. Motivated by this, the importance of economic aspects for wireless networks has been gradually noticed. As a consequence, many researchers started to study network economics by designing mechanisms to regulate the behaviors of participants [17]. Traditionally, studies on mechanism design in wireless communication networks, such as cognitive radio, concentrate on modeling situations where all participants attend a one-time decision. This implies that all participants are present when the mechanism starts, and that all participants wait for the mechanism's decision. This obviously omits an important characteristic of wireless communication networks, i.e., the mobility of mobile users. Since mobile users are not always stationary, they can opt-in and opt-out the network for their own convenience. This means that the demand of mobile users spans a period of time, which results in the fact that current decisions may affect future ones. Therefore, designing online incentive mechanisms to address such dynamics in wireless networks becomes pivotal and necessary.

Motivated by the above facts, in this thesis, online incentive mechanism designs are investigated in two different networks: crowdsensing network and edge computing networks. Concretely, in crowdsensing networks, an online incentive mechanism is proposed with random task arrival and QoI constraints. In edge computing networks, by considering dynamic task arrival in collaborative edge computing networks, an online incentive mechanism with computation resource allocation is designed. Moreover, a nonlinear online incentive mechanism is proposed to jointly optimize communication and computation resources in edge computing. To investigate the revenue of service providers, an incentive mechanism integrating machine learning techniques is designed in collaborative edge computing networks.

1.2.1 A Crowdsensing System

Mobile crowdsensing is a distributed problem-solving method where a crowd of mobile users (or participants) are dedicated to accomplishing a complicate task [18]. Today, a variety of applications based on crowdsensing have been developed to measure air/water quality in a specific area, ambient acoustic levels in a community, traffic congestion, parking availability, etc. [19–21].

There is a prevalent problem in mobile crowdsensing systems, which is to design an incentive mechanism to stimulate enough mobile devices to join the sensing campaign [22, 23]. Most recent works related to mechanisms in the literature can be categorized into offline and online cases. In the offline one, platform must know all information beforehand. For example, Luo et. al in [2] designed auction, lotteries, and trust and reputation systems while in [24], a randomized auction was proposed to overcome the reduced diversity in the deterministic auction mechanism. A more complex auction-based incentive mechanism by considering location information was investigated in [25]. To obtain higher quality data with less cost to the platform, a reverse combinatorial auction was designed in [26], which incorporated Quality of Information (QoI) in the mechanism design. In [27], authors designed a truthful incentive mechanism while taking the dependence of tasks' time window into account. In [28], the authors designed a quality-based incentive mechanism in which the designed payment method was closely related to the provision of users' QoI. While in the online incentive mechanism design, the platform should make decisions on whether or not taking this request only based on the known information so far. For example, authors in both [29] and [30] proposed the online incentive mechanisms to maximize the platform's utility. Since the first batch of users had to be used as the reference for afterward selection, this may cause more users be willing to arrive later because earlier arrivals had no benefits. Furthermore, in [31], two online mechanisms were proposed to maximize the platform's utility before the deadlines. These online mechanisms consisted of multiple-stage sampling-accepting processes, and at each stage, the platform allocated the tasks to the requestor, which was claimed in his bidding as long as its marginal price was no less than the precalculated threshold, and the budget allocated for each stage didn't deplete. A new paradigm for crowdsensing, called crowd foraging, was developed in [32], in which a self-organized mobile crowdsourcing model was achieved based on the opportunistic network and an optimal worker recruitment policy was proposed by using dynamic programming. Authors in [33] proposed a truthful online mechanism which aimed to minimize social cost throughout the system lifespan, while considering the users' location constraints and budget constraints when assigning sensing tasks. By applying the online learning techniques, authors in [34] proposed a quality-aware data online pricing mechanism, which aimed to minimize the expected regret under the budget constraints. Paper in [35] studied the task assignment and path planning problem in crowdsensing, and designed four online algorithms to maximize total task quality under constraints of users' budget.

Different from all aforementioned works, we consider another practical case in the mobile crowdsensing system. Particularly, in the considered mobile crowdsensing system, the following three practical factors are considered.

- Most existing work considered the arrival of mobile users in sequel. Actually, the tasks broadcasted by the platform can also arrive one by one. A practical application is illustrated in Fig. 2.1, where tasks for detecting parking availability arrive sequentially and randomly.

- In addition, the mobile crowdsensing system is aiming to obtain the sensed information as good as possible from various mobile users. Thus, in order to guarantee the quality of sensed data, the QoI for each task should be considered. Specifically, the QoI ensures that for any task that none of the single mobile user can satisfy the QoI demand, collective efforts of multiple mobile users are used to guarantee the sensing quality [26].

- Furthermore, each enrolled mobile users should consume their valuable resource, such as computation, communication, and energy resources once they are selected for executing crowdsensing tasks. Therefore, the designed online truthful mechanism must guarantee no violation on mobile users' cost budget, which is defined as the maximal value of the available resource mobile users can provide [36].

However, designing such online mechanism is challenging because i) In an online mechanism, the platform needs to make decisions as soon as possible without the presence of statistical information about future arriving tasks and future bidders. Thus, maintaining a good approximation ratio to the optimal solution based on incomplete information is very difficult; ii) The main purpose of our mechanism design is to guarantee the incentive compatibility so that each mobile user's actual bidding becomes the dominate strategy. In the offline mechanism design, such as Vickrey-Clarke-Groves (VCG) based [24] or Myerson based [37] mechanisms, the offline problem can be solved and the truthfulness can be guaranteed by using these outcomes. Without knowing future information, designing an online mechanism to keep truthfulness is extremely difficult than the offline mechanism design [38]; iii) The constraints of tasks' QoI constraints and each mobile user's cost budget induce a combination of typical covering and packing problems, which is NP-hard.

1.2.2 A Collaborative Edge Computing System

As the explosive development of smart devices and the advent of more and more new applications such as interactive gaming, video processing, and object recognition, traditional mobile cloud computing architectures become limited in meeting the requirements of these latency-critical applications due to potentially long propagation delay. Furthermore, improving network capacity along with higher data rates and lower latency is the essential requirement in the coming 5G networks [39]. To address these issues, Edge Computing (EC), firstly proposed by the European Telecommunications Standards Institute (ETSI) [40], is deemed as one of the key technologies for meeting these demands by deploying cloud servers at the edge of radio access networks. A recent trend of EC is to integrate collaborative offloading framework [41–44], where the computational tasks from requesters, such as mobile users and Ipads, can not only be executed by EC servers, but also be offloaded to nearby available mobile terminals called collaborators. As a complementary to EC systems, this collaborative offloading framework can achieve a win-win situation for both idle mobile terminals and network operators [45]. Recent studies have been done to investigate auction models for resource allocation in could computing systems, with quiet few of them [46–49] studying auction mechanisms for EC. Authors in [46] proposed a three-hierarchically architecture in EC, where an auction-based profit maximization problem was formulated. The work focused on maximizing the gained profit by jointly considering the revenue of serving the Virtual Machine (VM) demands, the electricity cost of running computing and network facilities, and the revenue lost due to network delay. A social welfare maximization problem was formulated in [47] to optimize continuous channel allocation under the constraint of a total number of channels and CPU resources. By revealing the competitions between different miners in blockchain network, the authors in [48] focused on the trading between the cloud/fog computing service provider and miners, and designed an auction mechanism to achieve optimal social welfare. Besides, a double auction model was proposed in [49], in which mobile devices were buyers and cloudlets were sellers. Note that all the aforementioned incentive mechanisms were designed for the non-cooperative task offloading and can only be used in a static setting (i.e., offline setting). In cooperative task offloading, [50] established a computation offloading reverse auction in vehicular networks where a VCG-based mechanism problem is formulated while satisfying the desirable economical properties of truthfulness and individual rationality.

Motivated by the above works, with the consideration of the following practical issues in collaborative EC systems, we aim to design an online incentive mechanism, which can not only encourage more participators, but also optimize both communication and computation resource allocation.

- In reality, collaborative offloading will cost storage, computation and communication resources of both EC servers and collaborators. From the economical perspective, EC servers have no responsibility to execute tasks from requesters without any reimbursement. In addition, idle mobile terminals are commonly intelligent and selfish [51, 52], and may not be willing to serve as collaborators. Thus, it is imperative to design an incentive mechanism, which can encourage both idle mobile users and EC servers to participate in collaborative offloading by offering them certain rewards for compensating their resource consumption.

- In practice, computation tasks do not arrive at the same time. Thus, it is infeasible for the central controller (e.g., the BS) to collect knowledge of all tasks before making offloading decisions, which makes the traditional offline solutions infeasible and requests the design of online algorithms.

- Practical computation tasks commonly have some tolerance in delay [45, 53]. Thus, suitably scheduling transmission and computation time periods within these tasks' delay tolerance becomes possible to make full use of network dynamics in channel conditions and computation resources, and to achieve diversity along the time.

Unfortunately, designing such mechanism is very challenging because i) In an online mechanism, the BS needs to make decisions right away for each arriving task without future task information. Therefore, maintaining a good performance compared to the optimal offline solution becomes very difficult; ii) The online mechanism needs to jointly optimize task executor selection, transmission and computation resources allocation, and transmission and computation time scheduling. Such joint optimization problem falls in a typical scope of combinatorial optimization and mixed integer programming, which is extremely difficult to solve; iii) The rational relationship between transmission and computation processes and the indetermination of transmission and computation resources allocation introduce nonlinear constraints, which further perplex the formulated optimization problem; iv) In a multiple-collaborator task offloading system, task information, such as data size, maximal execution delay, and preference, needs to be known to the BS for optimally managing the network. However, in practice, this information is private and unknown to the BS so that the selfish and intelligent requesters can intentionally report false information so as to maximize their own benefits. This requests that the designed online mechanism should not only incentivize both collaborators and the BS, but also prevent requesters from misreporting their information.

1.2.3 A Task Offloading System for Nonlinear Mechanism Designs

Task offloading in EC has attracted substantial attention from both industries and academia, where incentive mechanism designs in EC systems have gradually become a hot-discussed topic. In order to enable incentive mechanisms to be applicable in dynamic settings, [54–57] proposed online incentive mechanism designs for edge computing systems. Authors in [54] designed an online profit maximization multi-round auction for the computation resource trading between edge clouds (sellers) and mobile devices (buyers) in a competitive environment. For making full use of idle computation resources, online incentive mechanisms for collaborative task offloading in edge computing were proposed in [55, 56], where edge server motivated idle resourceful mobile users to provide their computation resources to requesters who would like to offload tasks and participate in the system dynamically. Moreover, by considering the energy harvesting process at mobile users, an optimization problem with a long-term reward objective was formulated in [57] to investigate sustainable computation offloading in an edge computing system, and then an online incentive mechanism based on the Lyapunov method and the VCG payment was designed to solve this problem. Even though the above-mentioned work were trying to design online incentive mechanisms in EC systems, their considered systems are linear, which means the objectives and constraints are all linear. However, in practice, it is required to design online incentive mechanisms with nonlinearity in both constraints and objectives. For example, when D2D technology is applied in EC networks to assist bad state mobile users for task offloading, interferences and power allocations in this network could be existed and considered, which makes the online incentive mechanism design problem nonlinear. In practice, these resourceful mobile users may not always be willing to actively execute tasks from IoT devices if no reimbursements are provided, as they need to consume their own computation and communication resources. Besides, computation tasks from IoT devices are generated along time, so that offloading decisions at the BS have to be made in an online manner without knowing information on possible future tasks. Moreover, it is well recognized that both communication and computation resources are limited at both the edge server and the mobile users. Therefore, in order to rationally utilize these limited resources and serve more tasks with their delay requirements, jointly optimizing communication resources, including transmission power and bandwidth for both upload link (from mobile users to the BS) and download link (from the BS to mobile users), and computation resources becomes mandatory. Most importantly, mobile users are battery-powered so that they are energy-constrained, or in other words, they have energy budgets. Without careful management, it would be possible that some of mobile users may use up their energy too fast to be available for any future participation. This may result in soaring maintenance cost as the remaining mobile users may ask for more reimbursements due to the reduction of competitions.

Motivated by the above facts, for a practical edge computing system, an incentive mechanism with the consideration of online decision making, joint computation and communication resource allocation, and energy budget has to be designed. However, designing such an online truthful mechanism is very challenging due to the following aspects. i) Joint consideration of transmission power, bandwidth, and computation resource allocation in both upload and download links introduces nonlinear constraints, and such a joint optimization problem belongs to typical combinatorial and mixed-integer programming. In addition, as aforementioned, tasks do not show up simultaneously in reality, leading to the design of an online algorithm. In combination, the considered problem falls into the scope of the nonlinear mixed-integer online optimization, which is extremely intractable; ii) The consideration of energy budget makes offloading decision-making coupled along the time, which prevents the simple solution of treating each time slot independently. Moreover, it would be difficult and non-trivial to keep a sound performance in comparison to the corresponding offline optimal solution while well balancing mobile users' energy consumption along the time.

1.2.4 A Task Offloading System for Revenue Maximization

Incentive mechanism design commonly needs to solve very complicated optimization problems. For example, for the objective of revenue maximization, the mechanism design has to jointly optimize resource allocation and pricing rule at the same time, and the formulated resource allocation problem is a multi-dimensional and nonlinear optimization one. This complicated optimization problem becomes even more difficult if a prior information on end users valuation is unknown. In the literature, most works, such as [46, 47, 49], focused on social welfare maximization, while few studies considered the revenue maximization due to its high complexity in solution [58]. However, the consideration of revenue maximization from the service provider's point of view is also important and meaningful because this reflects the benefit or profit that the service provider earns. Moreover, most existing work [59, 60] on revenue maximization mechanism design with the budget balance in wireless networks assumed the availability of *a priori* distributions of mobile users' private information. However, such an assumption may not always be true for practical EC applications. Furthermore, even though work in [61, 62] didn't need *a priori* distributions to design incentive mechanisms with budget balance, their objectives were actually not to maximize the revenue.

Artificial intelligence (AI) is envisioned as a promising technology to be employed in future wireless networks because of its flexibility and ability for solving large-scale problems [63]. AI is a broader concept of machine learning, which is probably the most popular application of AI. Moreover, it has been shown by many researches [64, 65] that compared to conventional methods, the utilization of machine learning can improve network performance. As such, a bunch of works [66–70] are starting to design computation resource allocation algorithms in edge computing networks based on machine learning techniques. For instance, a multi-user multi-edge-server computation offloading was formulated in [66] as a non-cooperative potential game where each mobile user maximized its obtained computation resource and reduced its energy consumption. This complex problem was solved by a model-free reinforcement learning technique. Through applying deep learning for data analysis in an edge computing system, paper in [67] designed a convolutional neural network to collaboratively perform image recognition between edge servers and the cloud in the proposed system. Furthermore, by integrating deep learning and reinforcement learning techniques, [68, 69] proposed deep reinforcement learning approaches in edge computing networks to maximize the long-term benefits of the whole system. Moreover, in [70], in order to obtain better offloading decisions, task execution time was firstly predicted by a low-rank learning model in the edge computing system. Then, the task offloading problem with predicted execution time was formulated to maximize the number of offloaded tasks.

Motivated by the above work, it is necessary to redesign revenue maximization incentive mechanisms by applying multi-task machine learning model with the consideration of unknown *a priori* distributions and at the same time satisfying some economical properties, such as individual rationality, incentive compatibility, and budget balance. However, such design is very challenging because of the following reasons. i) Traditionally, the designed mechanism should consist of an allocation rule and a pricing policy, where the allocation rule stipulates collaborator selection and the pricing policy determines how much mobile users need to pay. Unlike the social welfare maximization problem, where the allocation rule and the pricing policy can be designed independently, in the revenue maximization problem, these two aspects are tightly coupled, which greatly increases the difficulty in truthful mechanism design; ii) Without *a priori* distributions of mobile users' private information, the traditional solution methods for revenue maximization in the literature become infeasible. Therefore, a brand new approach has to be developed; iii)The net revenue of the service provider is defined as the sum of all payments from mobile users minus its costs. Thus, the designed mechanism needs to collect reimbursements as many as possible while still keeping the constraint of budget balance at each mobile user. These two requirements are contradictory with each other and further complex our problem; iv) In order to better assign collaborators, the designed mechanism should force mobile users to report their true private information. Meanwhile, mobile users who have tasks to offload may confront more than one idle collaborator available in their proximity, which results in a multi-dimensional bidding setting. Since most existing work on revenue maximization focused on just one- or two-dimensional bidding, those solutions cannot be directly applied to our case.

1.3 Summary of Contributions

The main objective of this thesis is to study economic aspects under dynamical settings in wireless networks by designing online incentive mechanisms. Specifically, two kinds of networks are taken into consideration, i.e., crowdsensing networks and edge computing networks. The main contributions are summarized as follows.

1) An Online Incentive Mechanism for Crowdsensing with Random Task Arrivals (this work has been published in IEEE Internet of Things Journal).

Different from existing online truthful mechanisms in crowdsensing system, for example [29–31], which considered the arrival of mobile users in sequel. However, in practice, the tasks broadcasted by the platform can also arrive one by one. Therefore, our work focuses more on on-the-fly tasks arrivals. In addition, existing work didn't consider the limited costs at mobile users because of the consumption of computation and energy resources. To this end, my designed online truthful mechanism must guarantee no violation on mobile users' cost budget. The main contributions of this chapter can be summarized as follows:

- A new payment method is designed and the convex decomposition technique is proposed for one-round auction problem.
- We prove our proposed online mechanism to be incentive-compatible, individual-rational, and be able to obtain a sound competitive ratio.
- Furthermore, an improved online truthful mechanism is proposed, which has a better performance if the platform gets some extra information *in prior*.
- Theoretical analyses on all properties of the proposed mechanisms have been provided, followed by comprehensive numerical simulations.

2) An Online Incentive Mechanism for Collaborative Task Offloading in Edge Computing (this work has been published in IEEE Transactions on Wireless Communications).

Most existing work [71, 72] focused on studying communication and computation resources allocation to minimize the system's energy consumptions or task delay, while ignoring the economical aspects. This is because mobile users and the EC server are selfish and have no responsibility to execute tasks for requesters if no rewards or reimbursement are provided. Thus, my work consider to design an incentive mechanism to encourage mobile users and the EC to participate. Moreover, a few existing work, such as [46–49], only considered the offline scenario to design incentive mechanisms for EC systems. My work considers a more practical case where computation tasks arrive along the time, and the design an online incentive mechanism. The main contributions of this chapter can be summarized as follows:

- An online incentive mechanism integrating task executor selection, resource allocation, and time scheduling is proposed for collaborative task offloading in a EC network. To the best of our knowledge, we are the first to jointly consider all these features for EC networks.
- We theoretically prove that the proposed online mechanism owns the properties of feasibility, computation efficiency with a competitive ratio of 3, incentive compatibility and individual rationality.
- Numerical simulations have been conducted to justify our theoretical analyses and verify the effectiveness of our proposed online mechanism.

3) Nonlinear Online Incentive Mechanism Design in Edge Computing Systems with Energy Budget (this work has been submitted to IEEE Transactions on Mobile Computing (Major Revision)).

In the literature, existing work, such as [71–74], mainly focused on communication or computation resource allocation under the assumption that mobile users and edge servers were willing to provide such computation service. Even though some researches [54–57] have studied the ways to incentivize mobile users or edge servers, energy budgets at mobile users were usually ignored and the more complicated resource allocation problem integrating power, bandwidth and computation resources, which is common for a practical system, has not been well addressed. Moreover, most of existing work considered offline incentive mechanism designs by assuming that all tasks arrived at the system at the same time. Different from all existing work, we consider a more challenging and realistic scenario in edge computing systems. The main contributions of this chapter can be summarized as follows:

- With the consideration of the energy budget at each mobile user, a nonlinear online incentive mechanism, integrating mobile user selection, communication resource allocation for both upload and download links, and computation resource allocation, is proposed for task offloading in the edge computing.

- In our proposed mechanism, a new framework for incentive mechanism design, called Integrate Rounding Scheme based Maximal-in-Distributional Range (MIDR) (IRSM), is proposed, which is applied to design the one-shot incentive mechanism.

- We theoretically prove that the proposed online mechanism has the properties of computation efficiency with a sound competitive ratio of $\beta(1 - \frac{1}{e})(2^{\phi} - 1)$, incentive compatibility and individual rationality.

- Numerical simulations have been conducted to justify our theoretical analyses and verify the effectiveness of our proposed online mechanism.

4) Truthful Deep Mechanism Design for Revenue-Maximization in Edge Computing with Budget Constraints (this work has been accepted in IEEE Transactions on Vehicular Technology).

In the literature, compared to works [46–49], which aimed to design incentive mechanisms by maximizing the social welfare of the system, fewer studies considered the revenue maximization due to its high complexity in solution [58]. However, the consideration of revenue maximization from the service provider's point of view is also important and meaningful because this reflects the benefit or profit that the service provider earns. Moreover, most existing works, such as [75–79], on revenue maximization mechanism design were under the constraint of Bayesian Incentive Compatibility (BIC), while we consider a Dominate-Strategy Incentive Compatibility (DSIC) mechanism design, which is a more stronger condition than BIC. In addition, some work [60, 80] on revenue maximization mechanism design in wireless networks assumed the availability of *a priori* distributions of smartphone users' private information. However, such an assumption may not always be true for practical EC applications. Furthermore, even though work in [61, 62] didn't need *a priori* distributions to design incentive mechanisms with budget balance, their objectives were actually not to maximize the revenue. Therefore, it is necessary to redesign revenue maximization incentive mechanisms by considering unknown *a priori* distributions and at the same time satisfying some economical properties, such as individual rationality, dominate-strategy incentive compatibility, and budget balance. The main contributions of this chapter can be summarized as follows:

- Different from the existing work which focused on social welfare maximization, our work focuses on the more challenging revenue maximization problem with budget balance. What's more, existing revenue maximization mechanisms can only be applied to a fixed bidding valuation distribution, while our proposed mechanism can be suitable for any distribution;
- Unlike [81], which considered one dimensional bidding setting and chose only one winner for each running time, a more general scenario where each mobile user can submit multiple biddings and multiple winners can be selected in each round is considered to design a truthful mechanism;
- Inspired by the multi-task machine learning model, a truthful deep mechanism is devised to fit our specific problem, and we further evaluate the performance of our proposed incentive mechanism through comprehensive numerical simulations.

1.4 Organization of the Thesis

The rest of the thesis is organized as follows. In Chapter 2, crowdsensing system is considered where the random task arrival with QoI guarantee is studied. In Chapter 3, a collaborative task offloading case is considered where an online incentive mechanism with computation resource allocation is investigated. By considering the nonlinearity in edge computing system, a nonlinear online incentive mechanism with budget constraints at mobile users is designed in Chapter 4. In Chapter 5, a new method by leveraging machine learning techniques to design incentive mechanisms in edge computing systems is presented, followed by conclusions and future work in Chapter 6.

Chapter 2

An Online Incentive Mechanism for Crowdsensing with Random Task Arrivals

In this chapter, we focus on random task arrivals and design an online truthful mechanism by jointly considering the cost budget and the requirement of sensed data of each participant. Specifically, we consider a scenario, as illustrated in Fig. 2.1, that there will be a concert in a certain area, which attracts a lot of people. Each participant will submit a task for checking the parking availability nearby specified by the concert sponsor, as the parking spaces are rather fewer. Clearly, in this scenario, tasks for inquiring parking availability arrive sequentially and randomly, and the sensing mobile users can be both recruited volunteers and employees maintaining parking orders and regulations in this area. To achieve this scenario, an online incentive mechanism is designed to minimize the social cost of the whole system and achieve truthfulness by applying the auction framework. Without considering constraints of cost budgets, an one-round incentive mechanism is designed through convex decomposition techniques, then we reconsider constraints of cost budgets and the designed one-round incentive mechanism to design an online incentive mechanism. Moreover, in order to further improve the competitive ratio of the online algorithm, a more efficient online scheme is proposed if more information on the participants is available at the platform. Theoretical and simulation results demonstrate the effectiveness of our proposed online truthful mechanisms.

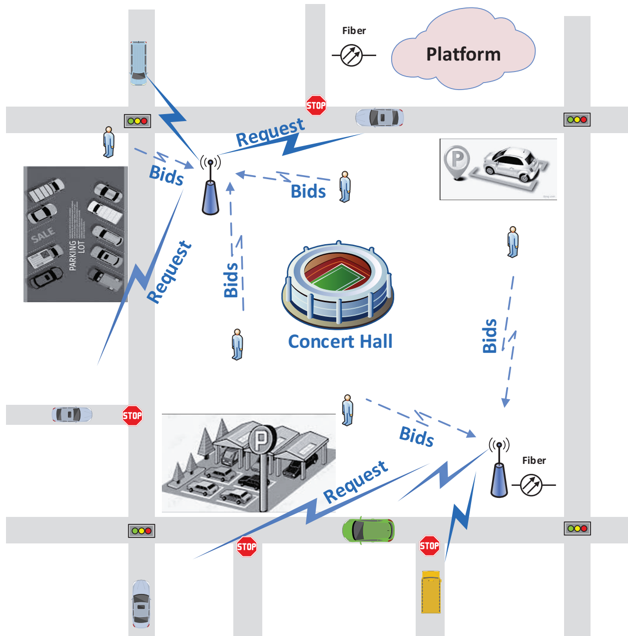

Figure 2.1: An illustrative scenario of crowdsensing.

2.1 System Model and Problem Formulation

In this section, the proposed system model is first described, and then an online auction is established to model the activities in this sensing campaign. After that the offline version of this online auction is formulated.

2.1.1 System model

We can summarize the aforementioned scenario as a crowdsensing system as shown in Fig. 2.2. The system includes a platform and mobile users (also called participants), denoted as a set $\mathcal{M} = \{1, 2, \cdots M\}$, who are willing to participate in crowdsensing[1]. For the feasibility of analysis, we consider a finite period of time ($\mathcal{T}$), and a bunch of tasks, denoted as $\mathcal{L} = \{1, 2, \cdots L\}$, arriving at the platform one by one during this period. Note that L is not necessary to be known to the

[1]The total number of mobile users are not fixed, and we can consider set $\mathcal{M}$ is large enough so that many mobile users' bids are empty. If some of the mobile users participate the system and submit bids later, the platform labels them and their bids.

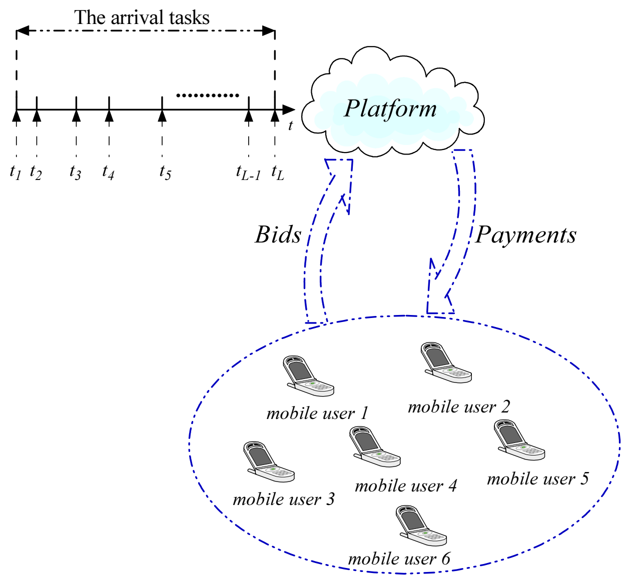

Figure 2.2: Crowdsensing system model.

platform in advance, and does not affect our online mechanism design. The definition of L is just used to facilitate the explanation of our system model and problem formulation. Denote the task arrival time instants as $\mathcal{T} = \{t_1, t_2, \cdots t_L\}, t_1 \leq t_2 \leq \cdots \leq t_L$, which is random and unknown to the platform in advance. The platform will announce the descriptions of tasks and recruit mobile users who are interested in participation. Note that mobile users have their cost budgets and the heterogenous requirements of the arrival tasks. Therefore, the described system can be modeled as a market, where the platform wants to buy the mobile users' services under certain regulations. As a consequence, an auction model becomes suitable to describe the activities between the platform and mobile users. More specifical, upon the arrival of each task, the bids including a cost[2], and the mobile user's cost budget are submitted to the platform immediately. The cost budget indicates the capacity limitation on service provision. As soon as all bids are collected, the platform has to select a fraction of these mobile users right away to process the current task and reimburse these chosen mobile users upon the completion. Even though we don't consider the mobility of recruited mobile users during the whole crowdsensing campaign, it has no effect on our proposed online

[2]Since each recruited mobile user should consume energy to execute the task and transmit the results, the cost is a compound of communication and transmission costs.

mechanism[3]. In this work, we assume that each mobile user submits N $(N \geq 1)$ bids for one task, and $v^{\ell}_{i,j}$ represents the real valuation of mobile user i's j^{th} bid. Collectively, we summarize the detailed procedures for this reverse auction as follows.

Upon receiving task ℓ, the platform and mobile users perform the following actions.

Step 1: The platform announces task descriptions to all the mobile users.

Step 2: A mobile user $i \in \mathcal{M}$ submits N bids for task ℓ, where each bid $b^{\ell}_{i,j}$, $j \in \mathcal{N}$, consists of a cost denoted by $c^{\ell}_{i,j}$ and its cost budget D_i. We denote all the bids sent from bidder i for the task ℓ as a set $\boldsymbol{B^{\ell}_i} = \{b^{\ell}_{i,j}, j \in \mathcal{N}\}$. Note that $b^{\ell}_{i,j}$ may not be the same as the actual $v^{\ell}_{i,j}$. In this chapter, we concentrate on designing a mechanism to force mobile users to bid the cost truthfully.

Step 3: The platform aims to maximize the social welfare of whole system under the QoI constraint for each task k and the cost budget of each mobile user, and chooses a subset of users for the task ℓ according to a well designed mechanism which will be discussed later.

Step 4: Upon the completion of a task, each winning mobile user i will obtain the payment p^{ℓ}_i. Note that p^{ℓ}_i is a function of user i's successful bid $b^{\ell}_{i,j}$.

Obviously, this reverse online auction is launched by the platform. Since the winning mobile user i will consume $c^{\ell}_{i,j}$ for the task ℓ and also obtain the reimbursement $p^{\ell}_i(b^{\ell}_{i,j})$, its utility equals $u_i(b^{\ell}_{i,j}) = p^{\ell}_i(b^{\ell}_{i,j}) - c^{\ell}_{i,j}$. Otherwise, we set $u_i(b^{\ell}_{i,j}) = 0$. Moreover, for notational clarity, we gather common used notations and their definitions in Table 2.1.

2.1.2 Problem Formulation

In this subsection, we first give definitions on some properties which are the desired goals for online auction design. Then an offline auction problem will be formulated.

Definition 8 (Individual Rationality (IR)). *A randomized auction mechanism [7] is individually rational if*

$$\mathbb{E}\{u_i(v^{\ell}_{i,j})\} \geq 0. \tag{5}$$

[3]This is because the movement of the recruited mobile users can only affect their sensing costs, and further impact their rewards from the platform. For instance, if one participator moves away from the concert center in Fig. 2.1, it will cost more to find an available parking space nearby concert center, which increases its sensing costs and decreases the winning probability in the competition.

Table 2.1: Commonly Used Notations In Chapter 2

Notation	Interpretation
$\mathcal{N}$	Set of mobile users
$\mathcal{L}$	Set of arrived tasks
t_i	Arriving time instant of task i
$\mathcal{N}$	Set of submitted bids
$b_{i,j}^{\ell}$	The j-th submitted bid of mobile user i for task ℓ
$c_{i,j}^{\ell}$	Cost of mobile user i for task ℓ on j-th bid
D_i	Cost budget for mobile user i
p_i^{ℓ}	The payment to i for task ℓ
u_i	utility of mobile user i
Q^{ℓ}	Required QoI for task ℓ
$a_{i,j}$, $d_{i,j}$	Starting and ending computation time instants
$w_{i,j}$	Modified cost of $c_{i,j}^{\ell}$
λ_k	The k-th weight in decomposition
$\boldsymbol{x}^k v_{i,j}$	The set of k-th integer solution in decomposition
$x_{i,j}^{\ell}$	Binary decision variable
$Q(\mathcal{B}^{\ell})$	The remained QoI to meet the requirement of task ℓ

Definition 9 (**Incentive Compatibility (IC)**). *A randomized auction mechanism is incentive-compatible if for any bid $b_{i,j}^{\ell} = v_{i,j}^{\ell}$, it is the dominant strategy. Let $b_{i,j}^{\ell'}$ denote the submitted bid (may be untruthful), and $\mathbf{b}_i^{\ell-}$ be the bid set without the bidder i. Then the incentive compatibility in expectation requires*

$$\mathbb{E}\{u_i(b_{i,j}^{\ell}, \mathbf{b}_i^{\ell-}) - u_i(b_{i,j}^{\ell'}, \mathbf{b}_i^{\ell-})\} \geq 0. \tag{6}$$

Definition 10 (**Computational Efficiency (CE)**). *An auction is computationally efficient if the time complexities of the algorithms applied in the auction are polynomial (i.e. the allocation determination and payment policy can be calculated in polynomial time).*

Definition 11 (**β-approximation Algorithm**). *For a given problem I and a given approximation algorithm $\mathcal{A}$, let $OPT(I)$ and $ALG(I)$ be the objective value of an optimal solution and the solution produced by $\mathcal{A}$, respectively. We call $\mathcal{A}$ as an β-approximation algorithm if*

$$\frac{ALG(I)}{OPT(I)} \leq \beta, \tag{7}$$

where $\beta > 1$ is called the approximation ratio.

For notation simplicity, we use p_i^ℓ to denote $p_i^\ell(b_{i,j}^\ell)$, and define a binary variable $x_{i,j}^\ell$. $x_{i,j}^\ell = 1$ means mobile user i's j-th bid for task ℓ wins. Otherwise, $x_{i,j}^\ell = 0$. Moreover, let $q_{i,j}^\ell$ be the corresponding achievable QoI for $b_{i,j}^\ell$. Note that the achievable QoI $q_{i,j}^\ell$ of each mobile user can be estimated by the platform through historical data[4] [82, 83], so that it is considered as a public information. Generally, the QoI refers to the quality of sensed data [26]. Define Q^ℓ be the QoI requirement for task ℓ, which also varies for different applications.

In this chapter, our objective is to design an auction mechanism that aims to minimize the total social cost (OFMSC) over all the tasks, which is defined as the summation of utilities from both mobile users and the platform. Since overall utilities of mobile users are $\sum\limits_{\ell\in\mathcal{L}}\sum\limits_{i\in\mathcal{M}} p_i^\ell - \sum\limits_{\ell\in\mathcal{L}}\sum\limits_{i\in\mathcal{M}}\sum\limits_{j\in\mathcal{N}} c_{i,j}^\ell x_{i,j}^\ell$ and the overall payment of platform is $\sum\limits_{\ell\in\mathcal{L}}\sum\limits_{i\in\mathcal{M}} p_i^\ell$, the social cost equals $\sum\limits_{\ell\in\mathcal{L}}\sum\limits_{i\in\mathcal{M}}\sum\limits_{j\in\mathcal{N}} c_{i,j}^\ell x_{i,j}^\ell$. Given all bidding information of mobile users, the QoI requirement of all tasks, and the cost budgets of all mobile users, the offline auction optimization problem is formulated as

$$\begin{aligned}
\text{OFMSC} \quad : \quad & \min_{\boldsymbol{X},\boldsymbol{P}} \sum_{\ell\in\mathcal{L}}\sum_{i\in\mathcal{M}}\sum_{j\in\mathcal{N}} c_{i,j}^\ell x_{i,j}^\ell \\
\text{s.t.}\ \ & C_1 : \sum_{j\in\mathcal{N}} x_{i,j}^\ell \le 1, \quad \forall\, \ell\in\mathcal{L}, \forall\, i\in\mathcal{M}; \\
& C_2 : \sum_{i\in\mathcal{M}}\sum_{j\in\mathcal{N}} q_{i,j}^\ell x_{i,j}^\ell \ge Q^\ell, \quad \forall\, \ell\in\mathcal{L}; \\
& C_3 : \sum_{\ell\in\mathcal{L}}\sum_{j\in\mathcal{N}} c_{i,j}^\ell x_{i,j}^\ell \le D_i, \quad \forall\, i\in\mathcal{M}; \\
& C_4 : \mathbb{E}\{u_i(b_{i,j}^\ell)\} \ge 0, \quad \forall\, i\in\mathcal{M},\ j\in\mathcal{N},\ \ell\in\mathcal{L}; \\
& C_5 : \mathbb{E}\{u_i(b_{i,j}^\ell, \boldsymbol{b}_i^{\ell-})\} \ge \mathbb{E}\{u_i(b_{i,j}^{\ell'}, \boldsymbol{b}_i^{\ell-})\},\ \forall\, i,\ j,\ \ell; \\
& C_6 : x_{i,j}^\ell \in \{0,1\},\ p_i^\ell \ge 0,\ \forall\, i\in\mathcal{M},\ j\in\mathcal{N},\ \ell\in\mathcal{L}.
\end{aligned}$$

where decision variables are $\boldsymbol{X} = \{x_{i,j}^\ell\}_{i\in\mathcal{M},\, j\in\mathcal{N},\, \ell\in\mathcal{L}}$ and $\boldsymbol{P} = \{p_i^\ell\}_{i\in\mathcal{M},\, \ell\in\mathcal{L}}$. In (OFMSC), constraint C_1 ensures at most one bid can be chosen by the platform for each task; C_2 is the QoI requirement for each task ℓ. Note that the same QoI model has also been adopted in [26], which

[4]Note that for any recruited mobile users who don't connect to the platform before, the platform can firstly estimate the QoI from their reported cost and sensing data [82].

means that for any task that none of the single mobile user can satisfy Q^ℓ so that collective efforts of multiple mobile users are used to guarantee the sensing quality; constraint C_3 guarantees that the consumed cost of each mobile user i must be no more than its cost budget; constraint C_4 is the individual-rational requirement; constraint C_5 ensures the incentive compatibility; constraint C_6 represents the allocation variables $x^\ell_{i,j}$ are binary, i.e. a mobile can either win or lose for each task ℓ, and the payment p^ℓ_i must be no less than zero for each mobile user i. Note that when the recruited mobile users can provide enough QoI for each task and not exceed their cost budgets, problem (OFMSC) always have a feasible solution of $\boldsymbol{X}$. In order to reduce the solution gap between problem (OFMSC) and its relax one [84], the constraint C_2 can be equivalently expressed as

$$C_2':\ \sum_{i\in\mathcal{M}}\sum_{j\in\mathcal{N}} q^\ell_{i,j}(\mathcal{B}^\ell)x^\ell_{i,j} \geq Q(\mathcal{B}^\ell),\ \forall\,(i,j)\notin\mathcal{B}^\ell,\ \mathcal{C}^\ell\subseteq\mathcal{C}^\ell,\ \ell\in\mathcal{L},$$

where $\mathcal{B}^\ell$ is the set of feasible mobile users who can meet the constraint C_2 for each task ℓ, and $\mathcal{C}^\ell$ denotes the set of all possible mobile users for the task ℓ; $Q(\mathcal{B}^\ell)$ and $q^\ell_{i,j}(\mathcal{B}^\ell)$ are the surplus QoI requirement and modified QoI given the current selected mobile users set $\mathcal{B}^\ell$, respectively. $Q(\mathcal{B}^\ell)$ and $q^\ell_{i,j}(\mathcal{B}^\ell)$ can be expressed as

$$Q(\mathcal{B}^\ell) = Q^\ell - \sum_{(i,j)\in\mathcal{B}^\ell} q^\ell_{i,j}, \tag{8}$$

$$q^\ell_{i,j}(\mathcal{B}^\ell) = \min\{q^\ell_{i,j},\ Q(\mathcal{B}^\ell)\}. \tag{9}$$

Note that the converted optimization problem is still NP-hard [85], even if the constraints C_3–C_5 are neglected in problem (OFMSC). Moreover, the platform cannot get the complete information *in prior*, so an online version of the problem (OFMSC) must be solved. To address these challenges, in the next section, a novel online auction mechanism is developed. Specifically, we first focus on designing the allocation rule and the pricing rule for each single arriving task ℓ, and then develop an online mechanism by jointly considering all tasks.

2.2 Online Mechanism Design

Since the pricing rule $\boldsymbol{P}$ is irrelevant to the objective of (OFMSC), we can first concentrate on designing an allocation rule by not considering the constraints C_4 and C_5, and then design a suitable pricing rule to satisfy both constraints C_4 and C_5. After removing constraints C_4 and C_5, the original optimization problem (OFMSC) can be rewritten as

P1
$$\min_{x_{i,j}^{\ell}} \sum_{\ell \in \mathcal{L}} \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} c_{i,j}^{\ell} x_{i,j}^{\ell}$$
$$\text{s.t. } C_1,\ C_2^{'},\ C_3,\ C_6.$$

2.2.1 Design of One-Round Auction

In this subsection, we will design an one-round auction algorithm. Note that, different from [7], our objective is to minimize the social cost so that the payment rule designed in [7] cannot be used in our case. Therefore, a novel payment rule will be developed to keep truthfulness. In the following, the superscript ℓ is omitted because of only single task under consideration. From (P1), the one-round optimization problem (P2) can be formulated as

P2
$$\min_{x_{i,j}} \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} w_{i,j} x_{i,j}$$
$$\text{s.t. } C_1,\ C_2^{'},\ C_6.$$

where $w_{i,j}$ is the modified cost of $c_{i,j}^{\ell}$, which is dependant on the remaining budget of mobile users. The calculation of $w_{i,j}$ will be introduced in subsection 2.2.2, and in this subsection, we can treat it as a constant. Note that in the (P2), we have intentionally removed the cost budget constraint C_3, which will be considered later.

Obviously, (P2) is an integer programming problem. To solve this problem, we first relax the binary variables to be real numbers taking values between 0 and 1 so that the fractional solution to the relaxed P2 can be obtained by applying the inner point method [86] or the ellipsoid method [87]. Then, we scale this fractional solution by β, which is the ratio between the solution of (P2) by the approximation algorithm $\mathcal{S}$ and its optimal one. Note that according to **Definition 4**, the algorithm $\mathcal{S}$ is an β-approximation algorithm, and we will detail the derivation of the factor β and this β-approximation algorithm (i.e., algorithm $\mathcal{S}$) in the subsection 2.2.1. In follows, we treat β as a constant temporarily and use the term β-approximation algorithm and algorithm $\mathcal{S}$ interchangeably. We decompose the scaled fractional solution into a polynomial number of integer solution sets and corresponding weights, which can be seen as the probabilities to choose these sets. Finally, we randomly select an integer solution according to the corresponding probability as the final solution of problem (P2). Since the decomposition of the scaled fractional solution is the major step in solving the relaxed P2, we discuss it in detail as follows.

Decomposition of the scaled fractional solution βx^*

According to [7], the main aim of this decomposition is to get the integer solutions of problem (P2) and the corresponding probabilities to select these solutions by decomposing $\boldsymbol{x}^*$, where $\boldsymbol{x}^*$ is the solution to the relaxed P2. Mathematically, the decomposition equality can be written as

$$\sum_{k\in\mathcal{K}} \lambda_k \boldsymbol{x}^k = \beta \boldsymbol{x}^*, \tag{10}$$

where λ_k and $\boldsymbol{x}^k$ denote the weight and the integer solution, respectively. It is worthy noting that the decomposition in (10) may not always be feasible. This is because the sum of the left-hand side of (10) must be less than 1, while some elements in the vector $\beta \boldsymbol{x}^*$ may be larger than 1 as the scaling factor β is no less than one[5]. In order to keep the decomposition feasible, we replace $\beta \boldsymbol{x}^*$ by $\min\{\beta \boldsymbol{x}^*, \mathbf{1}\}$, where $\mathbf{1}$ denotes an all "1" row vector with the same dimension as the vector $\boldsymbol{x}^*$.

[5]Since β is the solution ratio between problem (P2) and its relaxation, the ratio β must be larger than 1 [88].

To derive parameters λ_k and $\boldsymbol{x}^k$, the following linear optimization problem should be solved.

P3
$$\begin{aligned} &\max \sum_{k\in\mathcal{K}} \lambda_k \\ \text{s.t. } &\sum_{k\in\mathcal{K}} \lambda_k \boldsymbol{x}^k = \min\{\beta\boldsymbol{x}^*, \mathbf{1}\}; \\ &\sum_{k\in\mathcal{K}} \lambda_k \le 1; \quad \lambda_k \ge 0, \ \forall\, k \in \mathcal{K}, \end{aligned}$$

where $\mathcal{K}$ represents the set of all potential integer solutions of (P2). If the solution to (P3) $\hat{\lambda}_k$ satisfies $\sum_{k\in\mathcal{K}} \hat{\lambda}_k = 1$, it is our desired result and the decomposition is completed. Note that the problem (P3) is extremely difficult to be solved in a polynomial time due to the exponential number of variables. To address this problem, we consider its dual problem which can be expressed as

P4
$$\begin{aligned} &\min \sum_{(i,j)\in\mathcal{Y}} \min\{\beta x^*_{i,j}, 1\} h_{i,j} + H \\ \text{s.t. } &\sum_{(i,j)\in\mathcal{Y}} x^k_{i,j} h_{i,j} + H \ge 1, \ \forall\, k \in \mathcal{K}; \\ &h_{i,j} \text{ unconstrained}, \ H \ge 0, \end{aligned}$$

where $h_{i,j}$ and H are dual variables of problem (P3), and the set $\mathcal{Y}$ consists of all feasible fractional solutions of relaxed P2. Moreover, according to [86], it is easy to prove that the objective value between problems (P3) and (P4) is equal. Note that we can solve problem (P4) by the ellipsoid method polynomially, even if it has an exponential number of constraints.

Theorem 1. *For both optimization problems,* (P3) *and* (P4)*, their objective function equals 1 at optimum. Besides, both problems can be solved polynomially with a given algorithm $\mathcal{S}$.*

Proof. We first show that both problems (P3) and (P4) have the optimal objective of 1. From the feasible region of dual problem (P4), we can easily conclude that setting $\hat{h}_{i,j} = 0$ and $\hat{H} = 1$ is a feasible solution. Since minimization is the objective of dual problem, the minimum value of the objective function is at most 1. We assume that there exists another feasible solution $(\hat{h}_{i,j},\ \hat{H})$ such that

$$\sum_{(i,j)\in\mathcal{Y}} \min\{\beta x^*_{i,j}, 1\}\hat{h}_{i,j} + \hat{H} < 1.$$

According to the β-approximation algorithm, we can always find an integer solution $\bar{x}_{i,j}$, i.e. $\exists\, k \in \mathcal{K}$, satisfying $\sum\limits_{(i,j)\in\mathcal{Y}} \overline{x}_{i,j} h^+_{i,j} \le \sum\limits_{(i,j)\in\mathcal{Y}} \beta x^{**}_{i,j} h^+_{i,j} \le \sum\limits_{(i,j)\in\mathcal{Y}} \beta x^*_{i,j} h^+_{i,j}$, in which $x^{**}_{i,j}$ is the fractional optimal solution of problem (P2) with the input $h^+_{i,j}$. Moreover, we have the following inequality

$$\begin{aligned}
\sum_{(i,j)\in\mathcal{Y}} x^k_{i,j}\hat{h}_{i,j} + \hat{H} &= \sum_{\substack{\beta x^*_{i,j}>1 \\ or\ \hat{h}_{i,j}<0}} \hat{h}_{i,j} + \sum_{\substack{\beta x^*_{i,j}\le 1 \\ and\ \hat{h}_{i,j}\ge 0}} \overline{x}_{i,j} h^+_{i,j} + \hat{H} \\
&\le \sum_{\substack{\beta x^*_{i,j}>1 \\ or\ \hat{h}_{i,j}<0}} \min\{\beta x^*_{i,j}, 1\}\hat{h}_{i,j} + \sum_{\substack{\beta x^*_{i,j}\le 1 \\ and\ \hat{h}_{i,j}\ge 0}} \beta x^*_{i,j}\hat{h}_{i,j} + \hat{H} \\
&= \sum_{(i,j)\in\mathcal{Y}} \min\{\beta x^*_{i,j}, 1\}\hat{h}_{i,j} + \hat{H} < 1.
\end{aligned}$$

This contradicts the first constraint of dual problem (P4). Thus, the optimal solution of (P4) is 1. Since the strong duality meets, the primal problem (P3) also have the optimal solution of 1.

Next we prove the primal and dual problems can be solved in polynomial time with the β-approximation algorithm. Based on the previous discussions, we can add the following redundant constraint to problem (P4) without affecting optimality:

$$\sum_{(i,j)\in\mathcal{Y}} \min\{\beta x^*_{i,j}, 1\}h_{i,j} + H \le 1.$$

Recall that the dual problem (P4) has an exponential number of constraints but a polynomial number of variables. As a result, we can run the ellipsoid method on this new dual problem, and use the β-approximation algorithm as a separation oracle. The rule to construct the separation oracle is as follows. Assume that $(\hat{h}_{i,j},\ \hat{H})$ is the center of the current ellipsoid. $\sum\limits_{(i,j)\in\mathcal{Y}} \min\{\beta x^*_{i,j}, 1\}\hat{h}_{i,j} + \hat{H} \ge 1$ could be a separation oracle as long as $\sum\limits_{(i,j)\in\mathcal{Y}} \min\{\beta x^*_{i,j}, 1\}\hat{h}_{i,j} + \hat{H} \le 1$, otherwise, the designed β-approximation algorithm can be run to find a separation oracle. Thus, this new dual problem and the problem (P4) can be solved in polynomial time by the ellipsoid method. Since the number of constraints of problem (P4), which is solved by the ellipsoid method in polynomial time, is now polynomial, the number of both variables and constraints of its primal problem are also polynomial, which can be solved in polynomial time. This complete the proof for **Theorem** 1. ■

Theorem 1 guarantees the allocation rule of one-round auction to be computationally efficient if algorithm $\mathcal{S}$ is provided. In the following, we introduce the procedures of algorithm $\mathcal{S}$.

Derivation of scaled factor β and algorithm $\mathcal{S}$

Note that based on **Theorem** 1, to deal with both problems (P3) and (P4) polynomially, we need to establish a separation oracle [87] for P4. In the following, we derive the scaling factor β and design an β-approximation algorithm through the primal-dual method. We relax problem (P1) and let λ_i^ℓ, $\gamma(\mathcal{B}^\ell)$, μ_i be the dual variables related to constraints C_1, C_2', C_3, respectively. Then, the dual problem (P5) of the relaxed P1 can be formulated as

$$\text{P5} \qquad \max_{\substack{\mu_i,\lambda_i^\ell\\ \gamma(\mathcal{B}^\ell)}} \sum_{\substack{\ell\in\mathcal{L}\\ \mathcal{B}^\ell\subseteq\mathcal{B}^\ell}} \gamma(\mathcal{B}^\ell)Q(\mathcal{B}^\ell) - \sum_{i\in\mathcal{M}} D_i\mu_i - \sum_{\substack{\ell\in\mathcal{L}\\ i\in\mathcal{M}}} \lambda_i^\ell$$

$$\text{s.t. } C_7: \sum_{\mathcal{B}^\ell\subseteq\mathcal{C}^\ell:(i,j)\notin\mathcal{B}^\ell} \gamma(\mathcal{B}^\ell)q_{i,j}^\ell(\mathcal{B}^\ell) \le c_{i,j}^\ell + c_{i,j}^\ell\mu_i + \lambda_i^\ell,\ i\in\mathcal{M},\ j\in\mathcal{N},\ \ell\in\mathcal{L};$$

$$C_8: \lambda_i^\ell \ge 0, \mu_i \ge 0, \gamma(\mathcal{B}^\ell) \ge 0, i\in\mathcal{M},\ \ell\in\mathcal{L}.$$

From the problem (P5), the one-round dual problem (P6) of (P2) can be written as

$$\text{P6} \qquad \max_{\gamma(\mathcal{B}),\,\lambda_i} \sum_{\mathcal{B}\subseteq\mathcal{C}} \gamma(\mathcal{B})Q(\mathcal{B}) - \sum_{i\in\mathcal{M}} \lambda_i$$

$$\text{s.t. } C_9: \sum_{\mathcal{B}\subseteq\mathcal{C}:(i,j)\notin\mathcal{B}} \gamma(\mathcal{B})q_{i,j}(\mathcal{B}) \le w_{i,j} + \lambda_i,\ i\in\mathcal{M},\ j\in\mathcal{N};$$

$$C_{10}: \lambda_i \ge 0, \gamma(\mathcal{B}) \ge 0, i\in\mathcal{M}.$$

The basic idea of our β-approximation algorithm is to continuously greedily choose mobile users, i.e., solving primal variables and solve the dual variables based on the complementary slackness theorem [89]. This iteration will be continued if the constraint C_2' is met. The detailed procedure of this $\mathcal{S}$ algorithm is presented in **Algorithm** 1. In **Algorithm** 1, line 1 to line 6 initialize settings. The first condition of line 7 ensures that this algorithm can have a finite number of iterations, while the violation of the second condition means the feasible solution has been found. Line 9 is the greedy method to choose the mobile users and the dual variables are calculated in line 10. Finally, the dual variables λ_i and $\gamma(\mathcal{A})$ are fitted in line 14 and line 15, respectively.

Algorithm 1: Algorithm $\mathcal{S}$.

1 **Initialization**;
2 $\gamma(\mathcal{B})$=0, $\forall \mathcal{B} \subset \mathcal{C}$;
3 $x_{i,j} = 0, \ \forall i \in \mathcal{M}, \ \forall j \in \mathcal{N}.$ ▷primal variables;
4 $\mathcal{B} = \emptyset$;
5 $\mathcal{R}^{sel} = \emptyset$;
6 $\hat{w}_{i,j} = w_{i,j}, \ \forall i, j$;
7 **while** $\mathcal{M} \neq \emptyset$ ***and*** $Q(\mathcal{B}) > 0$ **do**
8 $\quad \hat{w}_{i,j} = \hat{w}_{i,j} - \gamma(\mathcal{B}) q_{i,j}(\mathcal{B}) \ \forall \, i, j$;
9 $\quad (i^*, j_{i^*}) = \underset{i \in \mathcal{M}/\mathcal{R}^{sel}}{\arg\min} \{ \frac{\hat{w}_{i,j}}{q_{i,j}(\mathcal{B})} \}$;
10 $\quad \gamma(\mathcal{B}) = \frac{\hat{w}_{i^*, j_{i^*}}}{q_{i^*, j_{i^*}}(\mathcal{B})}$;
11 $\quad x_{i^*, j_{i^*}} = 1, \ \ \mathcal{B} = \mathcal{B} \bigcup (i^*, j_{i^*}), \ \mathcal{R}^{sel} = \mathcal{R}^{sel} \bigcup i^*$;
12 $\quad \mathcal{M} = \mathcal{M}/i^*$;
13 **Dual fitting** ;
14 $\lambda_i = 0, \ \forall i \in \mathcal{M}$; ▷ variables in constraint C_1 ;
15 $\alpha = \underset{(i_1, i_2) \in \mathcal{M}, \, (j_1, j_2) \in \mathcal{N}}{\max} \{ \frac{w_{i_1, j_1}}{w_{i_2, j_2}}, \ \frac{w_{i_1, j_1} q_{i_2, j_2}}{w_{i_2, j_2} q_{i_1, j_1}} \}$;
16 $\gamma(\mathcal{B}) = \gamma(\mathcal{B})/\alpha$;

Lemma 1. *After the termination of the while loop,* ***Algorithm*** *1 can produce a feasible solution to the P2.*

Proof. In **Algorithm** 1, a new mobile user is added to the set $\mathcal{B}$ in each loop while $Q(\mathcal{B}) > 0$. Once the while loop terminates, we have $Q(\mathcal{B}) \leq 0$, which means the algorithm has found a feasible solution satisfying constraint C_2'. In addition, since line 9 only chooses at most one bid from each bidder i, the constraint C_1 can also be guaranteed. Moreover, line 11 sets the deciding variables $x_{i,j}$ to 1 which is initiated by 0, so that the constraint C_4 is met. Although the **Algorithm** 1 may also stop due to $\mathcal{M} = \emptyset$, which means we can not select enough mobile users to satisfy the constraint

C_2', this case may hardly happen in practice when there are a large number of mobile users. Thus, the **Algorithm** 1 will output a feasible solution to the P2. ■

Lemma 2. ***Algorithm** 1 generates a feasible solution to the dual problem* (P6).

Proof. Please refer to Appendix A.1. ■

Theorem 2. ***Algorithm** 1 is an β-approximation algorithm with $\beta = 2\alpha$ and $\alpha \geq 1$, where α is the dual fitting factor.*

Proof. Please refer to Appendix A.2. ■

Note that the dual variable $h_{i,j}$ in (P4) may be negative, while algorithm $\mathcal{S}$ needs the inputs to be positive. To address this issue, we introduce $h_{i,j}^+$ as the input of **Algorithm** 1, which is defined as

$$h_{i,j}^+ = \begin{cases} \hat{h}_{i,j} & \text{if } \hat{h}_{i,j} \geq 0 \quad \text{and} \quad \beta x_{i,j}^* \leq 1; \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

We denote $\overline{x}_{i,j}$ as the integer solution obtained by **Algorithm** 1 with the input $h_{i,j}^+$. Then the integer solution for the decomposition can be modified accordingly as

$$x_{i,j}^k = \begin{cases} \overline{x}_{i,j} & \text{if } \hat{h}_{i,j} \geq 0 \quad \text{and} \quad \beta x_{i,j}^* \leq 1; \\ 1 & \text{otherwise.} \end{cases} \tag{12}$$

Till now, the whole allocation rule of this one-round auction is completed. In the following, we design the payment rule for this one-round auction that ensures constraints C_4 and C_5.

Design of payment rule

In this chapter, we design a new payment rule for the one-round auction. Note that the fractional VCG rule [24] that has been successfully used in [7, 90] to guarantee truthfulness in expectation is not applicable in our case[6]. Our designed payment for winning user i can be expressed as

$$p_i = \frac{\sum\limits_{\substack{v=i \\ s=v+1}}^{I+1} c_{s,j_s}(f_v(c_{v,j_v}) - f_s(c_{s,j_s}))}{f_i(c_{i,j_i})}, \ (i,j_i) \in \mathcal{Y}, \tag{13}$$

where $I = |\mathcal{Y}|$, $f_i(c_{i,j})$ is the winning probability of mobile user i in the one-round auction, and $\boldsymbol{c}_{i,j}^{-}$ is the set of other bids excluding mobile user i.

Theorem 3. *The designed one-round auction is truthful, individual-rational, and β-approximation in expectation.*

Proof. we first prove the incentive compatibility by showing the following three statements.

1) Since the decision variables $x_{i,j}^k$ is binary, we have $\mathbb{E}\{x_{i,j}^k\} = f_i(c_{i,j})\times 1+(1-f_i(c_{i,j}))\times 0 = f_i(c_{i,j})$, and $f_i(c_{i,j}) = \mathbb{E}\{x_{i,j}^k\} = \sum\limits_{k\in\mathcal{K}} \lambda_l x_{i,j}^k = \min\{\beta x_{i,j}^*, 1\}$. For the fractional solution of problem (P2), since the increment of $c_{i,j}$ can lead to the increase of modified cost $w_{i,j}$, which results in the non-increasing of corresponding $x_{i,j}^*$ because of the minimization in the objective of (P2), $\min\{\beta x_{i,j}^*, 1\}$ is also non-increasing in $c_{i,j}$. Thus, $f_i(c_{i,j})$ is monotonically non-increasing with respect to $c_{i,j}$.

2) Since any mobile user has its own budget on its bids, $x_{i,j}$ will be zero once user i has ran out of its cost budget. Thus, we have $\int_0^\infty f_i(\boldsymbol{c})d\boldsymbol{c} = \int_0^{D_i} f_i(\boldsymbol{c})d\boldsymbol{c} < \infty, \ \forall i$.

3) In the auction, winners are paid based on the threshold payment. Here, the expected threshold payment can be calculated as follows.

After calculating the fractional solution of problem (P2), we can obtain the winning probability $f_i(c_{i,j})$ of each user. Then, we sort these winning probabilities in a non-increasing order, and the corresponding cost $c_{i,j}$ in the non-decreasing order as

$$f_1(c_{1,j_1}) \geq f_2(c_{2,j_2}) \geq \cdots \geq f_v(c_{v,j_v}) \geq \cdots \geq f_I(c_{I,j_I}),$$
$$c_{1,j_1} \leq c_{2,j_2} \leq \cdots \leq c_{v,j_v} \leq \cdots \leq c_{I,j_I}.$$

[6]This is because the objective of the problems in [7, 90] is maximization, which leads to the satisfaction of the equation ($\sum\limits_{k\in\mathcal{K}} \lambda_k \boldsymbol{x}^k = \beta \boldsymbol{x}^*$), while in this chapter, we have inequality ($\sum\limits_{k\in\mathcal{K}} \lambda_k \boldsymbol{x}^k \leq \beta \boldsymbol{x}^*$) because the minimization is considered. Therefore, a truthful mechanism based on VCG rule cannot be held in our case.

Note that we have rearranged the indexes. Thus, the expected threshold payment for user i is

$$\mathbb{E}\{p_i\} = \sum_{\substack{v=i \\ s=v+1}}^{I+1} c_{s,j_s}(f_v(c_{v,j_v}) - f_s(c_{s,j_s})),\ (i, j_i) \in \mathcal{Y}. \tag{14}$$

Since $\mathbb{E}\{p_i\} = f_i \times p_i + (1-f_i) \times 0$, the payment of winning user is $p_i = \frac{\sum_{\substack{v=i \\ s=v+1}}^{I+1} c_{s,j_s}(f_v(c_{v,j_v}) - f_s(c_{s,j_s}))}{f_i(c_{i,j_i})}$.

In summary, the first and second statements ensure $f_i(c_{i,j})$ is monotonically non-increasing in $c_{i,j}$ and bounded, respectively, while the last statement indicates (14) is the expected threshold payment for each mobile user i. According to [37, 91], the designed randomized auction is truthful.

We now examine the individual rationality. For notational simplicity, let c_s and f_v denote c_{s,j_s} and $f_v(c_{v,j_v})$, respectively. Moreover, we let $f_v - f_{v+1} = \delta_v \geq 0$ and $c_{v+1} = c_i + \gamma_v \geq 0$, $v \in \{i, I+1\}$. Then, the expected utility of mobile user i is calculated as

$$\begin{aligned}
\mathbb{E}\{u_i\} &= \sum_{v=i}^{I+1} c_{v+1}(f_v - f_{v+1}) - f_i c_i = \sum_{v=i}^{I+1} (c_i + \gamma_v)\delta_v - f_i c_i \\
&= \sum_{v=i}^{I+1} \gamma_v \delta_v + f_i c_i - f_i c_i = \sum_{v=i}^{I+1} \gamma_v \delta_v \geq 0.
\end{aligned}$$

Thus, by using our designed price rule, the one-round auction is individual-rational.

At last, we prove this one-round auction is an β-approximation algorithm in expectation. Let E_{SC} denote the expected social cost. For truthful biding, we have

$$\begin{aligned}
E_{SC} &= \sum_{k \in \mathcal{K}} \lambda_k \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} w_{i,j} x_{i,j}^k = \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} w_{i,j} \sum_{k \in \mathcal{K}} \lambda_k x_{i,j}^k \\
&= \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} \min\{\beta x_{i,j}^*, 1\} w_{i,j} \leq \sum_{i \in \mathcal{M}} \sum_{j \in \mathcal{N}} \beta x_{i,j}^* w_{i,j} \\
&= \beta p_f^* \leq \beta p^*.
\end{aligned}$$

The last inequality holds because $\min\{\beta x_{i,j}^*, 1\} \leq \beta x_{i,j}^*$. Therefore, our one-round auction is an α-approximation algorithm which has the same approximation ratio as that of **Algorithm** 1. This completes the proof for **Theorem** 3. ■

For better understanding the whole process of our one-round auction design, we summarize the procedure in **Algorithm** 2.

Algorithm 2: One-round Auction.

Input: The modified cost $w_{i,j}$ and cost budget D_i.

Output: The allocation result (λ_k, $\boldsymbol{x}^k$), and the corresponding payment p_i.

Obtain the solution $\boldsymbol{x}^*$ by solving the relaxation problem (P2);

Decompose $\beta\boldsymbol{x}^*$ ($\beta \geq 1$) based on $\sum\limits_{k\in\mathcal{K}} \lambda_k \boldsymbol{x}^k = \min\{\beta\boldsymbol{x}^*, 1\}$;

Choose $\boldsymbol{x}^k$ with probability λ_k, $\forall k \in \mathcal{K}$;

For each mobile user i, the reimbursement $p_i = \frac{\sum\limits_{\substack{v=i\\ s=v+1}}^{I+1} c_{s,j_s}(f_v(c_{v,j_v}) - f_s(c_{s,j_s}))}{f_i(c_{i,j_i})}$, $(i, j_i) \in \mathcal{Y}$.

2.2.2 Design of Online Mechanism

Based on the designed one-round auction for one task, in this subsection, we design an online auction by considering a sequence of randomly arrived tasks, which can still guarantee a good CR to the corresponding offline solution.

In our online auction, the main factor under consideration is the residual cost budget which has been ignored in the one-round auction. The our designing rationality is that some of the cost-effective mobile users will be forbidden to join the future auction campaign, if they use up their cost budget too early. This will result in the increase of the social cost. Based on this observation, our proposed online truthful auction increase the modified costs of the mobile users once they have been chosen before, so as to decrease the winning probability, and extend their lifetime in the whole auction process. The procedure of this truthful online auction is summarized in **Algorithm** 3. Line 2 is the initial process of dual variables $\mu_i^{(\ell)}$ which are associated with cost budget constraints. In line 3, we use the "for loop" to describe the arriving tasks, and for each arrived task, the modified costs in line 4 are calculated based on the real cost of the previous winning mobile users. In line 5, a subset of winners can be obtained by running **Algorithm** 2. Both lines 6 and 7 update the dual variables for each winning mobile user. Hereby, $\phi = \max\limits_{\substack{i\in\mathcal{M},\, j\in\mathcal{N}\\ \ell\in\mathcal{L}}} \{D_i/c_{i,j}^{\ell}\}$, which can be obtained from historical information.

To better understand our online mechanism, we use an example to illustrate the procedure.

Table 2.2: The Costs of Two Mobile Users

	Task 1	Task 2	Task 3
Actual cost (user 1, user 2)	(5, 6)	(6, 7)	(7, 10)
Modified cost (user 1, user 2)	(5, 6)	(8, 7)	(8, 12)

Suppose that there are three tasks arriving in sequence and two mobile users. The actual costs and the modified costs of each mobile user for these three tasks are shown in Table 2.2, and we assume the cost budgets are 11 and 13 for mobile users 1 and 2, respectively. We compare the following two cases.

Case I: selection based on the actual cost of mobile users;

Case II: selection based on the modified cost of mobile users (i.e., by using the **Algorithm** 3).

For case I, it will choose mobile user 1 for tasks 1 and 2, and mobile user 2 for task 3. This is because the cost budget of mobile user 1 has almost depleted after task 2, and can not bid for task 3. The total cost is 5+6+10=21. For case II, the proposed mechanism will choose the mobile user 1 for tasks 1 and 3, and mobile user 2 for task 2, which results in the overall cost is 5+7+7=19<21. Note that by adopting this online strategy, the proposed online mechanism can produce similar results of the problem (OFMSC) as that of using optimization method. This is because the **Algorithm** 2 can solve the problem (OFMSC) sub-optimally in the one-round auction without considering budget constraint, i.e., C_3. Moreover, the proposed online mechanism strategically allocates cost budgets of each mobile user to different arriving tasks, which can further improve the performance, as illustrated in the above example.

Algorithm 3: The online truthful mechanism.

$\mu_i^{(0)} = 0,\ i \in \mathcal{M}$;

for *each task* ℓ **do**

  $w_{i,j} = c_{i,j}^{\ell} + c_{i,j}^{\ell}\mu_i^{(\ell-1)}$, $\triangleright$ update the modified cost;

  Execute **Algorithm** 2 to get the set of winning mobile users, which is denoted as $\mathcal{G}_k$, and j_i is the selected bid index of mobile user i;

  $\mu_i^{(\ell)} = \mu_i^{(\ell-1)}(1 + \frac{c_{i,j_i}^{\ell}}{\beta D_i}) + \frac{c_{i,j_i}^{\ell}}{\beta \phi D_i},\ i \in \mathcal{G}_k$;

  $\mu_i^{(\ell)} = \mu_i^{(\ell-1)},\ i \in \mathcal{M}/\mathcal{G}_k$;

Lemma 3. *The **Algorithm** 3 can produce feasible solutions for both problems* (P1) *and* (P5).

Proof. Since **Algorithm** 2 (line 2) applies **Algorithm** 1 to obtain the feasible integer solution $x_{i,j}^k$, the output of **Algorithm** 3 is feasible for each task ℓ. As a consequence, $x_{i,j}^k$ satisfies constraints C_1, $C_2^{'}$, and C_4. Since it is rational that each mobile user will not submit the bids for tasks that exceed its cost budget, constraint C_3 can also be met for each mobile user i for sure. Collectively, **Algorithm** 3 can produce feasible solutions to problem (P1). We continue to prove the feasibility of problem (P5). From line 4 of **Algorithm** 3, we have $w_{i,j} = c_{i,j}^{\ell} + c_{i,j}^{\ell}\mu_i^{(\ell-1)}$. By substituting this equality into constraints C_9 of problem (P6), we have

$$\begin{aligned}\sum_{\mathcal{B}\subseteq\mathcal{C}:(i,j)\notin\mathcal{B}} \gamma(\mathcal{B})q_{i,j}(\mathcal{B}) &\leqslant c_{i,j}^{\ell} + c_{i,j}^{\ell}\mu_i^{(\ell-1)} + \lambda_i \\ &\leqslant c_{i,j}^{\ell} + c_{i,j}^{\ell}\mu_i + \lambda_i,\end{aligned}$$

where $\mu_i = \mu_i^{(L)}$ and L is the index of the last arrived task. The last inequality holds because of the non-decreasing property of μ_i. This completes the proof. ∎

Theorem 4. *The proposed online auction with a CR of $\beta\frac{\phi}{\phi-1}$ is truthful and individual rational in expectation.*

Proof. Since IC and IR can be directly derived from **Theorem** 3, we focus on proving that the competitive ratio of the **Algorithm** 3 is $\beta\frac{\phi}{\phi-1}$. Denote $P^{(\ell)}$ and $D^{(\ell)}$ as the objective value after the task ℓ has been finished in problem (P1) and its dual problem (P5), respectively. Then, we have

$$\begin{aligned}
\Delta P^{(\ell)} &= \sum_{i\in\mathcal{G}_k} c_{i,j}^{\ell} = \sum_{i\in\mathcal{G}_k}(w_{i,j_i} - c_{i,j_i}^{\ell}\mu_i^{(\ell-1)}) \\
&= p - \sum_{i\in\mathcal{G}_k} c_{i,j_i}^{\ell}\mu_i^{(\ell-1)} \\
&= p - \beta\sum_{i\in\mathcal{G}_k} D_i(\mu_i^{(\ell)} - \mu_i^{(\ell-1)}) + \frac{\sum_{i\in\mathcal{G}_k} c_{i,j_i}^{\ell}}{\phi} \\
&\le \beta d - \beta\sum_{i\in\mathcal{G}_k} D_i(\mu_i^{(\ell)} - \mu_i^{(\ell-1)}) + \frac{\Delta P^{(\ell)}}{\phi} \\
&= \beta\Delta D^{(\ell)} + \frac{\Delta P^{(\ell)}}{\phi}, \\
\Rightarrow \Delta P^{(\ell)} &\le \beta\Delta D^{(\ell)} + \frac{\Delta P^{(\ell)}}{\phi} \le \beta\frac{\phi}{\phi-1}\Delta D^{(\ell)},
\end{aligned}$$

where $\Delta P^{(\ell)} = P^{(\ell)} - P^{(\ell-1)}$ and $\Delta D^{(\ell)} = D^{(\ell)} - D^{(\ell-1)}$. By considering $\Delta P^{(0)} = \Delta D^{(0)} = 0$, $P^{(L)} \le \beta\frac{\phi}{\phi-1}D^{(L)}$ is held. Therefore, the CR of this online truthful auction is $\beta\frac{\phi}{\phi-1}$. ■

Remarks: If the parameter α happens to be 1, which means all the mobile users are homogenous, the competitive ratio equals $2\frac{\phi}{\phi-1}$.

2.2.3 Improved Online Mechanism Framework

Note that even though the online strategy is effective in reducing the overall cost, it may suffer from many unused budgets of the cost-efficient mobile users. Thus, in this subsection, we try to further improve the performance of the proposed online mechanism. The improvement is based on the observation that the dual variable μ_i is updated exponentially in **Algorithm** 3, which may decrease the winning probability of cost-efficient mobile users too fast for the future tasks. Now if the platform knows that each mobile user would like to spend at least a fraction η_i, $(0 < \eta_i \le 1)$ of its budget over the whole tasks, we can define a suitable threshold μ_i^d so that when $\mu_i^{(\ell)} = \mu_i^d$, user i has just used $\eta_i D_i$ amount of its budget. Then, if $\mu_i^{(\ell)} \le \mu_i^d$, $\mu_i^{(\ell)}$ is updated linearly, and otherwise, updated exponentially. The newly designed online mechanism is shown in **Algorithm** 4.

Algorithm 4: The improved online truthful mechanism.

$\mu_i^{(0)} = 0,;$

$\mu_i^d = \frac{\eta_i ln\phi}{\phi^{(1-\eta_i)\beta-1}}, i \in \mathcal{M}$;

for *each arriving task* ℓ **do**

  $w_{i,j} = c_{i,j}^{\ell} + \beta c_{i,j}^{\ell} \max\{\mu_i^{(\ell-1)}, \mu_i^d\}$;

  Execute **Algorithm** 2 to get the set of winning mobile users ;

  $\mu_i^{(\ell)} = \mu_i^{(\ell-1)} + (\max\{\mu_i^{(\ell-1)}, \mu_i^d\} + \frac{\mu_i^d(1-\eta_i)}{\eta_i})\frac{c_{i,j}^{\ell}}{D_i},\ i \in \mathcal{Y}_k$;

  $\mu_i^{(\ell)} = \mu_i^{(\ell-1)},\ i \in \mathcal{M}/\mathcal{Y}_k$;

Lemma 4. *Let* $\hat{\eta} = \min\limits_{i \in \mathcal{M}} \eta_i$. *The competitive ratio of the improved online mechanism equals* $\frac{\beta\phi^{(1-\hat{\eta})\beta+1}}{\phi^{(1-\hat{\eta})\beta+1}-(1-\hat{\eta})\beta ln\phi}$, *which is better than* ***Algorithm*** *3.*

Proof. Since the proof procedure is similar to that for Theorem 4, we only present the main steps. According to **Algorithm** 4, we have

$$\begin{aligned}
\Delta P^{(\ell)} &= \sum_{i \in \mathcal{Y}_k} (w_{i,j_i} - \beta c_{i,j}^{\ell} \max\{\mu_i^{(\ell-1)}, \mu_i^d\}) \\
&\leq \beta d - \beta \sum_{i \in \mathcal{Y}_k} D_i(\mu_i^{(\ell)} - \mu_i^{(\ell-1)}) + \frac{\beta u_i^d(1-\eta_i)}{\eta_i}\Delta P^{(\ell)} \\
&= \beta \Delta D^{(\ell)} + \frac{\beta u_i^d(1-\eta_i)}{\eta_i}\Delta P^{(\ell)}.
\end{aligned}$$

Thus, by substituting u_i^d into above inequality, we have

$$\begin{aligned}
\Delta P^{(\ell)} &\leq \frac{\eta_i \beta}{\eta_i - \alpha u_i^d(1-\eta_i)}\Delta D^{(\ell)} \\
&= \frac{\beta\phi^{\beta(1-\eta_i)+1}}{\phi^{\beta(1-\eta_i)+1} - \beta(1-\eta_i)ln\phi}\Delta D^{(\ell)} \\
&\leq \frac{\beta\phi^{(1-\hat{\eta})\beta+1}}{\phi^{(1-\hat{\eta})\beta+1} - (1-\hat{\eta})\beta ln\phi}\Delta D^{(\ell)}.
\end{aligned} \tag{15}$$

The last inequality in (15) holds because the competitive ratio is a convex function with respect to η_i, and reaches the maximum at $\hat{\eta}$. Finally, we compare the competitive ratios obtained by

Algorithm 3 and **Algorithm** 4. From (15), we get

$$\frac{\beta\phi^{(1-\hat{\eta})\beta+1}}{\phi^{(1-\hat{\eta})\beta+1}-(1-\hat{\eta})\beta ln\phi}=\beta(1+\frac{\beta(1-\hat{\eta})}{\frac{\phi}{ln\phi}\phi^{(1-\hat{\eta})\beta}-(1-\hat{\eta})\beta})$$
$$\leq\beta(1+\frac{\beta(1-\hat{\eta})}{\phi\beta(1-\hat{\eta})-\beta(1-\hat{\eta})})=\beta(1+\frac{1}{\phi-1})=\beta\frac{\phi}{\phi-1},$$

where the inequality follows $\frac{\phi}{ln\phi}\phi^{(1-\hat{\eta})\beta}\geq\phi\beta(1-\hat{\eta})$. It means the competitive ratio of **Algorithm** 4 is better than **Algorithm** 3. ■

Next, we begin to evaluate the computation complexity of our proposed algorithm, and use the same metric as in [92, 93]. The proposed online mechanism consists of two parts, i.e., one-round auction, and **Algorithm** 3. In the one-round auction, the inner point method is used to solve the relaxed problem (P2), which has the computational complexity $O((NM)^{3.5}ML)$. After that, the decomposition technique is applied to this fractional solution, and has computational complexity $O((NM)^6(ML)^2)$. Thus, the total computational complexity is $O((NM)^{3.5}ML+(NM)^6(ML)^2)$ for each one-round auction. Given there are L tasks arriving at the platform in total, the overall computational complexity is $O(((NM)^{3.5}ML+(NM)^6(ML)^2)L)$.

2.3 Numerical Results

In this section, we evaluate the performance of our proposed online truthful auction by performing numerical simulations in Matlab. In the simulation, similar to [24, 26, 94], we assume the uniform distributions for random variables. Specifically, the real cost, the estimated QoI, and the cost budget of each user are chosen uniformly from (2, 4), (1, 2), and (6, 25), respectively. Since the proposed auction algorithms do not relay on specifical distribution, a similar observations can be obtained for other distributions. We obtain the estimated parameter $\hat{\alpha}$ by selecting the largest α among all the tasks, and 600 different runs are used averagely for one point in the follows figures.

Fig. 2.3 reveals the optimality of one-round and online auctions when the number submitted bids of mobile users and arrival tasks are $N = 2$ and $L = 15$, respectively. As shown in the figure, the social cost value decreases with the increase of registered mobile users. The reason is that platform has more potential "cheaper" mobile users to choose. Besides, approximation ratio

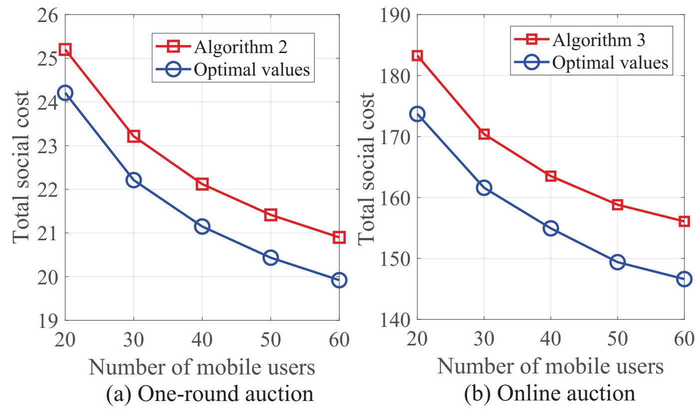

Figure 2.3: Number of mobile users v.s. Total social cost.

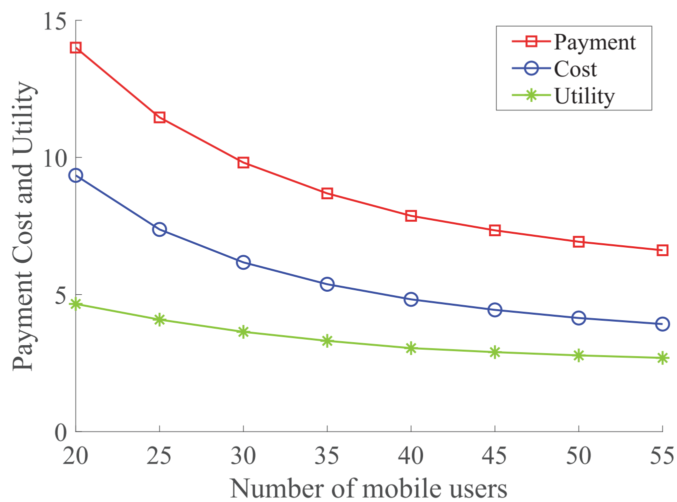

Figure 2.4: Number of mobile users v.s. Average payment, Cost, and Utility.

obtained by one-round auction in Fig. 2.3 (a) is a bit better than that in Fig. 2.3 (b). This is because only a single task is considered in one-round auction, while in the online auction, multiple tasks are considered without *a prior* information. Moreover, the calculated values based on **Algorithm** 2 and **Algorithm** 3 are just a little bit higher than the corresponding optimal ones. The figure also demonstrates that online to offline ratio of our proposed online algorithm matches our theoretical analysis (For instance, usually $\beta \approx 6.5$ and $\phi \approx 14$ in our setting, so that the worst case competitive ratio can be about 6.5).

Fig. 2.4 evaluates the average payment, the biding cost, and the utility of the recruited mobile users with the increasing number of mobile users when $L = 15$, $N = 2$. As we can see that the

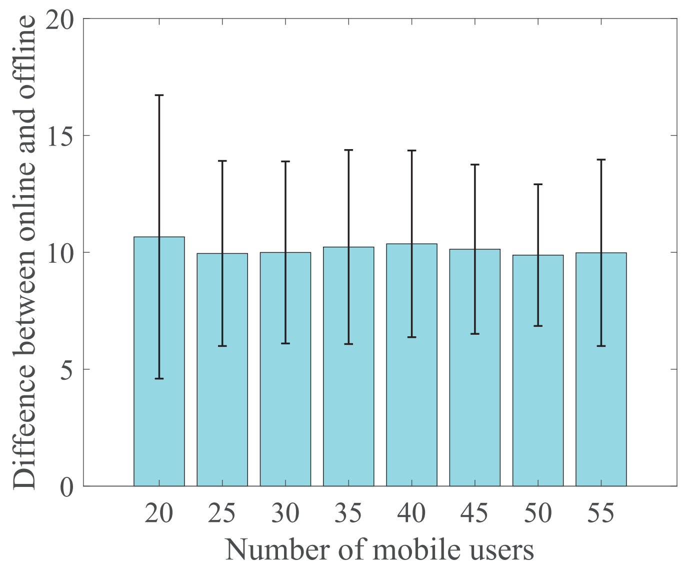

Figure 2.5: Number of mobile users v.s. The difference between online and offline.

average payment and the utility are decreasing when the number of mobile users are increasing. This is because the winning probability of each mobile user decreases with the number of mobile users. Therefore, the payments received from the platform are gradually decreasing, and so are the utilities. Moreover, this figure also shows that the payment to mobile user is larger than the cost so that the utility obtained by mobile user also is always larger than zero, which justifies the individual rationality of our proposed online auction. Under the same simulation conditions as in the Fig. 2.4, Fig. 2.5 shows the difference of objective function between offline and online algorithms (i.e., **Algorithm** 3). From this figure, it can be observed that the difference is almost steady and the maximal variance is almost 6, which further demonstrate the robustness of our proposed online mechanism.

Fig. 2.6 shows the relationship between the cost budget of mobile users and the total social cost with $M = 40$, $N = 2$, and $L = 15$. From Fig. 2.6, we can observe that when the cost budget is small, the total social cost is relatively high. In addition, the total social cost decreases with the increase of the cost budget till saturation. This can be explained as follows. When the cost budget is very small, the cost-efficient mobile users can easily run out their budgets, which forces the platform to choose mobile users with higher costs. With the rise of their cost budgets, more cost-efficient bids can be provided by mobile users so that the total social cost is decreased. Furthermore,

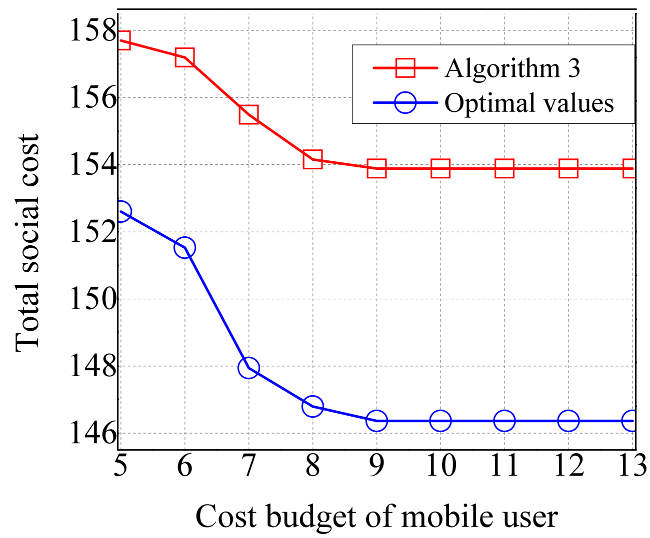

Figure 2.6: Cost budget v.s. Total social cost.

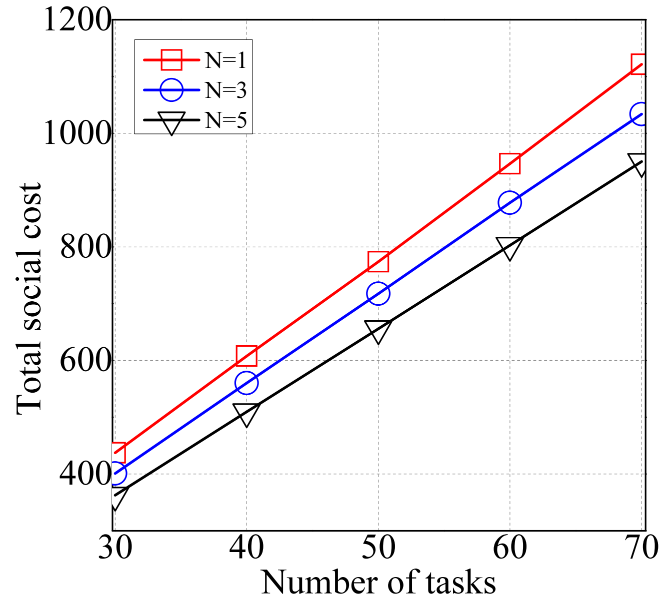

Figure 2.7: Number of arriving tasks v.s. Total social cost.

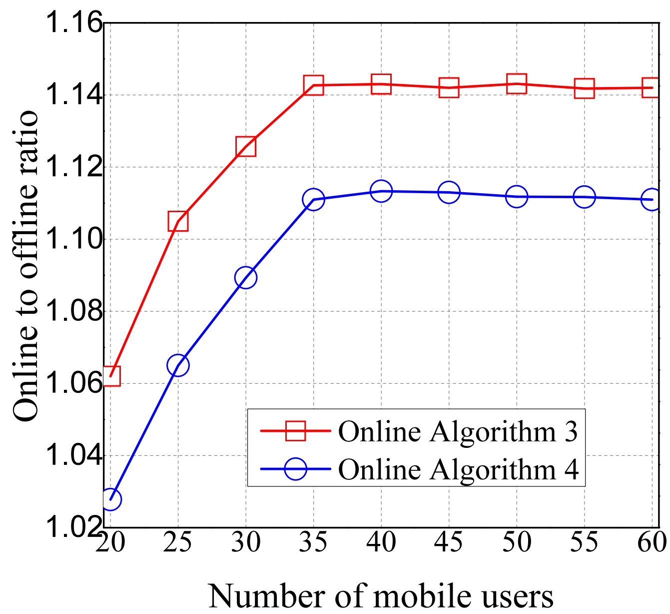

Figure 2.8: The effect of number of mobile users on online to offline ratio.

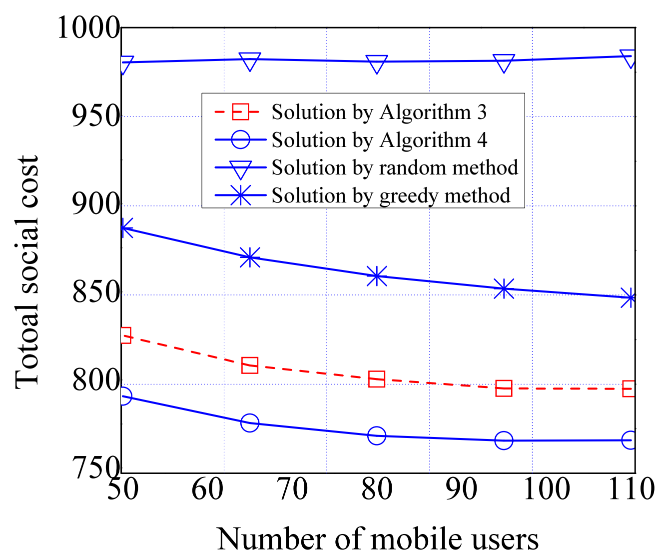

Figure 2.9: Number of mobile users v.s. Total social cost.

when the cost budget is large enough, since there are always cost-efficient mobile users to meet the QoI requirements of all tasks, no higher cost mobile users will be selected.

Fig. 2.7 illustrates the effects of overall arrived tasks on total social cost. We consider different bidding numbers for each user (i.e., $N = 1$, $N = 3$, $N = 5$), and set $M = 100$. We can see that the value of total social cost is rising with more tasks arriving at the platform. This is obvious since the objective function in problem (OFMSC) sums all of the arrived tasks, the total social cost surely increase with more arrived tasks. Furthermore, the total social cost will decrease if multiple bids are submitted for a single task. This is because the platform can treat each bid as a virtual mobile user. Then, with the increase of the number of bids from each mobile user, the number of potential virtual mobile users becomes larger, which increases the possibility to find more cost-efficient users in crowdsensing.

Fig. 2.8 shows the effect of the number of mobile users in terms of the online to offline ratio derived by both **Algorithm** 3 and **Algorithm** 4 when $N = 2$, $L = 56$. From Fig. 2.8, we observe that the online to offline ratio first increases and then becomes saturated after $M = 35$. This is because the total social cost decreases with the increase number of mobile users and the offline result will reduce faster due to its optimality. However, when there are sufficient number of mobile users, since the platform has enough cost-efficient mobile users to select, the ratio becomes stable. Moreover, the online to offline ratio by **Algorithm** 4 is lower than that by **Algorithm** 3, which clearly demonstrates the effectiveness of our improved online algorithm.

In Fig. 2.9, the proposed online **Algorithm** 3 and **Algorithm** 4 are compared with other two online algorithms, i.e., greedy algorithm and random algorithm, in terms of the total social cost. In the greedy algorithm, the algorithm chooses the mobile users with the smallest cost until the QoI requirement for each task ℓ is satisfied, and the random online algorithm randomly chooses the mobile users till the satisfaction of QoI requirement for each task. In our simulation, we set $L = 56$ and $N = 2$. From Fig. 2.9, it is observed that the proposed algorithms outperform both the greedy algorithm and the random algorithm, and **Algorithm** 4 is even better than **Algorithm** 3. The reasons behind are that no information is considered in the random algorithm, while only "cheaper" bidings are taken into consideration in greedy algorithm, while ignoring the corresponding QoI information.

2.4 Summary

In this chapter, two online incentive mechanisms for mobile crowdsensing systems are proposed. By jointly considering QoI requirement and cost budget of mobile users, an online auction optimization problem is formulated as a minimization of social cost. To solve this problem and maintain truthfulness and individual rationality, we first design an one-round auction for a single task, and then expand the results to design online incentive mechanisms. Both theoretical and numerical results show that the proposed online mechanisms can ensure effectiveness and performance guarantee.

Chapter 3

An Online Incentive Mechanism for Collaborative Task Offloading in Edge Computing

In this chapter, incentive mechanism designs for collaborative task offloading in EC is studied. Different from most existing work in the literature that was based on offline settings, an efficient online incentive mechanism for collaborative task offloading in EC systems is proposed. In the considered system model, upon the arrival of a requester, it submits its private information to the central controller (i.e., the BS) to request a task offloading. After receiving the request, the BS makes decisions right away on task executor selection, time scheduling, resource allocation, and reward determination. With the objective of maximizing the total social welfare (the summation of utilities of all requesters and task executors), we formulate a complex optimization problem and design an online incentive mechanism based on the primal-dual framework. Specifically, we first convert the optimization problem to its dual form, and then by observing the dual constraints and its corresponding dual variables, we design two marginal price functions which are updated according to the current availability of resources. Based on these marginal price functions, we can decide the best task executor which has the maximal utility, and determine the optimal time scheduling and

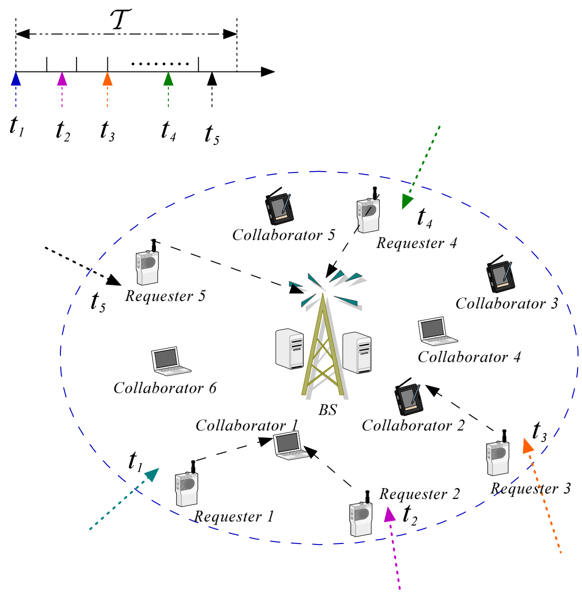

Figure 3.1: The system model of collaborative task offloading in edge computing where there are 5 requesters arriving at the network in an online fashion and submitting their requests.

corresponding resource allocation. Finally, theoretical analyses show that our mechanism can guarantee feasibility, truthfulness, and computational efficiency (competitive ratio of 3). We further use comprehensive simulations to validate our analyses and the properties of our proposed mechanism.

3.1 System model and Problem formulation

In this section, we describe the system under consideration, and model the interaction between the BS and arrived requesters as an online auction. After that, the corresponding offline optimization problem is formulated.

3.1.1 System model

We consider a edge computing network as shown in Fig. 3.1. A similar system has been discussed in [41, 45]. The system consists of a BS integrating edge servers and several mobile users who can also provide computation services, called collaborators. These collaborators are recruited by the BS and are willing to provide computation resources if reimbursements are given. Time to time, there are mobile users, called requesters, who request computing services. The requesters randomly arrive in a sequence, and we denote t_i as the arrival time instant of requester i. Note that the BS does not have any *a priori* information on requesters' arrival times.

Consider a time-slotted structure with a slot length of Δt. For each arrived requester $i \in \mathcal{U}$, where $\mathcal{U}$ denotes the set of all requesters, let $\mathcal{M}_i$ be the set of available collaborators that can provide computation services to it. Note that the set $\mathcal{M}_i$ could be available to requester i by applying the discovery approach [95]. We further define $\mathcal{N}_i = \mathcal{M}_i \cup 0$, where the index 0 represents the BS. Obviously, $\mathcal{N}_i$ consists of all collaborators and the BS that requester i can offload its task to. For the notation simplification, we will use the term "task executor" to denote any collaborator or the BS throughout this chapter. We further let $\mathcal{M}$ be the set of all collaborators. Note that since user mobility cannot affect the offloading process in the coverage of one EC server, we don't consider it in this chapter. This is because if the collaborator moves away after the arrival of the requester's task, the task will fail to be transmitted to the collaborator, and no rewards can be obtained. So, there are no incentives for collaborators to move. For the movement of requester, if the requester moves around the corresponding collaborator, D2D technologies [96] can be applied to transmit the computational results and the reimbursements between requester and collaborator. Otherwise, results and reimbursements can also be received and transmitted through the cellular link.

The task from requester $i \in \mathcal{U}$ is denoted as $T_i = (s_i, \tau_i)$, where s_i is the size (in bits) of the offloaded task and τ_i is the maximal tolerance delay. Note that the task offloading to the collaborator can be done through a D2D link [96]. Each task i requires Q_i CPU cycles for execution and can be calculated by $Q_i = \kappa_i s_i$ [71] where κ_i is the CPU cycles coefficient. We also define the allocated CPU frequency at task executor j as $f_{i,j}$. Then, the required computation time at task executor j for task i equals

$$I_{i,j}^C = \frac{Q_i}{f_{i,j}} = d_{i,j} - a_{i,j}, \tag{16}$$

where $a_{i,j}$ and $d_{i,j}$ denote the starting and ending computation time instants, respectively. In addition, given that each requester i is allocated an orthogonal channel with bandwidth $\phi_{i,j}$ for task offloading to task executor j, the transmission rate from requester i to task executor j equals

$$r_{i,j} = \phi_{i,j} \log_2(1 + \gamma_{i,j}), \tag{17}$$

where $\gamma_{i,j} = \frac{\Lambda_{i,j}|h_{i,j}|^2}{\sigma^2}$ is the Signal-to-Noise Ratio (SNR), σ^2 is the average power of background noise, and $\Lambda_{i,j}$ and $h_{i,j}$ are the transmission power and the channel gain between requester i and task executor j, respectively. Thus, the transmission time from requester i to task executor j can be calculated as

$$I^T_{i,j} = \frac{s_i}{r_{i,j}} = o_{i,j} - g_{i,j}, \tag{18}$$

where $g_{i,j}$ and $o_{i,j}$ denote the starting and ending transmission time instants, respectively. Note that since both the available computation and transmission resources are time-varying, both transmission time and computation time should be optimally determined for each offloading task. Therefore, $g_{i,j}$, $o_{i,j}$, $a_{i,j}$, and $d_{i,j}$ are decision variables. To meet the task delay requirement, we need

$$d_{i,j} - t_i \leq \tau_i. \tag{19}$$

In (19), similar to studies in [97,98], we ignore the time for the task executor to send the computation result back to the requester because the data size of outcomes for many applications is commonly very small. In summary, the whole operation procedure of this system is described as follows.

Step 1. Upon the arrival of requester i, it submits multiple bids to the BS, denoted by $B_{i,j} = (T_i, t_i, v_{i,j}), j \in \mathcal{N}_i$, where $v_{i,j}$ is the valuation of requester i to task executor j, which represents its preference to offload the task to task executor j [1].

Step 2. After collecting the bids from requester i, the BS makes a decision, denoted by a binary variable $x_{i,j}$, whether to accept this requester. $x_{i,j} = 1$ means requester i is accepted. Otherwise, $x_{i,j} = 0$. The BS further determines what are the optimal transmission and computation time instants for this task.

[1] Since the collaborators are heterogeneous in terms of available computation resources and geographical locations, which makes the channel conditions between the collaborators and the requesters different, each requestor values the nearby collaborators differently.

Step 3. The BS sends the optimal results obtained in Step 2 to requester i and notifies the selected task executor to prepare for the task execution.

Step 4. After the task is completed, requester i will be charged by $p_{i,j}$, which is another decision variable, and the task executor returns computation results to it.

Obviously, the interactions between task executors and requesters can be model as an online auction, where the BS is the auctioneer, requesters are buyers, and all the task executors are sellers. Requesters may strategically misreport their private information (i.e., $B_{i,j}$) in order to get more benefits. For example, requester i, who will lose in the auction, may submit false bid $B^{'}_{i,j}$, where $T^{'}_{i} = T_i, t^{'}_{i} = t_i$, and $v^{'}_{i,j} > v_{i,j}$. In this case, this requester have higher chance to win the auction than reporting truthfully. Thus, a truthful incentive mechanism is necessary for our considered system. Following the previous discussions, the utilities of requester i and the task executor j can be respectively expressed as

$$u_i = v_{i,j} - p_{i,j}, \tag{20}$$

$$u_j = p_{i,j} - e_{i,j}c_j, \tag{21}$$

where $e_{i,j} = Q_i \xi_j f^2_{i,j}$ and c_j are the energy consumption for executing the task i and the unit energy cost of task executor j, respectively, and ξ_j is the energy consumption coefficient [99]. For notational clarity, the commonly used abbreviations and notations in chapter 3 are summarized in Table 3.1.

Problem Formulation

Our target is to design an online auction which can satisfy the following properties.

- IC, which means no requesters can gain more utilities by misreporting their bids;
- IR, which guarantees utilities of all requesters are no less than zero;
- SW maximization. Here, SW is defined as a summation of all participators' utilities, and can

Table 3.1: Commonly Used Notations In Chapter 3

Notation	Interpretation
t_i	Arriving time instant
Δt	A time slot length
$\mathcal{M}_i$	Set of available collaborators of requester i
$\mathcal{U}$	Set of all requesters
$\mathcal{N}_i$	Set includes $\mathcal{M}_i$ and the BS
τ_i	Maximal tolerance delay
s_i	Size of task
Q_i	Required CPU cycles for task i
$f_{i,j}$	Allocated CPU frequency for task i task executor j
$I_{i,j}^C$	Required computation time at task executor j for task i
$a_{i,j}$, $d_{i,j}$	Starting and ending computation time instants
$I_{i,j}^T$	Required transmission time from i to j
$g_{i,j}$, $o_{i,j}$	Starting and ending transmission time instants
$v_{i,j}$	Valuation of requester i to j
$x_{i,j}$	Binary variable
$p_{i,j}$	Payment from requester i to j
u_i	Utility of requester i
$e_{i,j}$	Energy consumption at task executor j
c_j	Unit cost per energy
$\ell_{i,j}^1$ and $\ell_{i,j}^2$	Sets of all the feasible transmission and computation time scheduling

be calculated as

$$SW = \sum_{i \in \mathcal{U}} \sum_{j \in \mathcal{N}_i} w_{i,j} x_{i,j}, \tag{22}$$

where $w_{i,j} = v_{i,j} - e_{i,j} c_j$.

- Computational Efficiency (CE), which means the designed online mechanism should be run in polynomial time.

If the information about all tasks is known, we can formulate the corresponding offline optimization problem (MSW) as

$$
\begin{aligned}
\max_{\boldsymbol{X},\boldsymbol{G},\boldsymbol{O},\boldsymbol{A},\boldsymbol{D},\boldsymbol{P}} \ & \sum_{i\in\mathcal{U}}\sum_{j\in\mathcal{N}_i} w_{i,j}x_{i,j} \qquad\qquad \text{MSW}\\
\text{s.t.}\ \ & C_1: \sum_{j\in\mathcal{N}_i} x_{i,j} \le 1, \quad \forall\, i\in\mathcal{U};\\
& C_2: \sum_{j\in\mathcal{N}_i} (o_{i,j}-a_{i,j})x_{i,j} \le 0, \quad \forall\, i\in\mathcal{U};\\
& C_3: (16),\ (18) \text{ and } (19);\\
& C_4: \sum_{\substack{i\in\mathcal{U}:\\ a_{i,j}\le t\le d_{i,j}}} f_{i,j}x_{i,j} \le F_j, \quad \forall\, t\in\mathcal{T},\ j\in\mathcal{M}\cup\text{BS};\\
& C_5: \sum_{\substack{i\in\mathcal{U}:\\ a_{i,j}\le t\le d_{i,j}}} s_i x_{i,j} \le S_j, \quad \forall\, t\in\mathcal{T},\ j\in\mathcal{M}\cup\text{BS};\\
& C_6: \sum_{\substack{i\in\mathcal{U}:\\ g_{i,j}\le t\le o_{i,j}}} \sum_{j\in\mathcal{N}_i} \phi_{i,j}x_{i,j} \le W, \quad \forall\, t\in\mathcal{T};\\
& C_7: \sum_{j\in\mathcal{N}_i} (v_{i,j}-p_{i,j})x_{i,j} \ge 0, \quad \forall\ i\in\mathcal{U};\\
& C_8: \sum_{j\in\mathcal{N}_i} (v_{i,j}-p_{i,j})x_{i,j} \ge \sum_{j\in\mathcal{N}_i} (\widetilde{v}_{i,j}-\widetilde{p}_{i,j})x_{i,j}, \forall\ i\in\mathcal{U};\\
& C_9: x_{i,j}\in\{0,1\},\ o_{i,j}\in\{t_i,\tau_i\},\ g_{i,j}\in\{t_i,\tau_i\},\\
& \quad\ a_{i,j}\in\{t_i,\tau_i\},\ d_{i,j}\in\{t_i,\tau_i\},\ \forall\, i\in\mathcal{U},\ j\in\mathcal{N}_i,
\end{aligned}
$$

where $w_{i,j} = v_{i,j} - e_{i,j}c_j$, F_j and S_j denote the maximal CPU frequency and storage capacity of the executor j, respectively, W is the whole bandwidth of the system, and $\mathcal{T}$ is the set of all time slots. Decision variables are $\boldsymbol{X} = \{x_{i,j}\}_{i\in\mathcal{U},j\in\mathcal{N}_i}$, $\boldsymbol{G} = \{g_{i,j}\}_{i\in\mathcal{U},j\in\mathcal{N}_i}$, $\boldsymbol{O} = \{o_{i,j}\}_{i\in\mathcal{U},j\in\mathcal{N}_i}$, $\boldsymbol{A} = \{a_{i,j}\}_{i\in\mathcal{U},j\in\mathcal{N}_i}$, $\boldsymbol{D} = \{d_{i,j}\}_{i\in\mathcal{U},j\in\mathcal{N}_i}$, and $\boldsymbol{P} = \{p_{i,j}\}_{i\in\mathcal{U},j\in\mathcal{N}_i}$. Constraint C_1 ensures that each requester can offload its task to at most one task executor. Constraint C_2 means the transmission process occurs before the computation process for any task. Constraints C_3 represents the time rationality and delay requirement. Constraints C_4 and C_5 indicate constrains on the allocated CPU frequencies and storage resources at any task executor, respectively. Constraint C_6 specifies that the allocated bandwidths cannot exceed W, and constraints C_7 and C_8 are the requirements of IR and IC, respectively. Constraint C_9 defines decision variables $\boldsymbol{G}$, $\boldsymbol{O}$, $\boldsymbol{A}$, $\boldsymbol{D}$, and $\boldsymbol{P}$ to be continuous and $\boldsymbol{X}$ to be binary variables.

Obviously, this formulated offline optimization problem is a mixed integer problem and is usually NP-hard [100]. In addition, this formulation requires a complete information on system operation in the future. In the follows, we are going to design a novel online mechanism to find solutions on the fly.

3.2 Online Mechanism Designs

In this section, we try to design an online mechanism to find solutions to the problem (MSW). Note that since the payments are not in the objective function in (MSW) but only in the constraints C_7 and C_8, we can decouple (MSW) into two subproblems without losing optimality: an allocation subproblem (including task executor selection, resource allocation, and time scheduling) and a payment rule subproblem. In the following, we first reformulate the offline problem, and then solve the allocation problem. After that a corresponding payment scheme will be designed to not only satisfy IC, but also maintain IR.

3.2.1 Problem Reformulation

Since constraints C_4 and C_5 in (MSW) have the same structure, we combine them together as

$$C_{10}: \sum_{\substack{i \in \mathcal{U}: \\ a_{i,j} \le t \le d_{i,j}}} r_{i,j}^k x_{i,j} \le R_j^k, \quad \forall\, t \in \mathcal{T},\ \forall\, j \in \mathcal{M},\ \forall\, k \in \mathcal{K},$$

where

$$r_{i,j}^k = \begin{cases} s_i & \text{if } j \in \mathcal{N}_i \text{ and } k = 1; \\ f_{i,j} & \text{if } j \in \mathcal{N}_i \text{ and } k = 2, \\ 0 & \text{otherwise;} \end{cases} \qquad R_j^k = \begin{cases} S_j & \text{if } j \in \mathcal{M} \text{ and } k = 1; \\ F_j & \text{if } j \in \mathcal{M} \text{ and } k = 2; \\ 0 & \text{otherwise;} \end{cases}$$

$\mathcal{K} = \{1, 2\}$, and $l_{i,j} = l^1_{i,j} \cup l^2_{i,j}$ denotes all feasible transmission and computation time scheduling pairs from requester i to task executor j with satisfaction of constraints C_2, C_3, and C_9. Let $l^1_{i,j} = \{l^1_{i,j}(1), l^1_{i,j}(2), \cdots\}$ and $l^2_{i,j} = \{l^1_{i,j}(1), l^2_{i,j}(2), \cdots\}$ are sets of all the feasible transmission and computation time scheduling, respectively, and each entry $l^1_{i,j}(\ell)$ or $l^2_{i,j}(\ell)$ indicates the ℓ-th feasible scheduling scheme. Let $\mathcal{L}_{i,j}$ be the index set of all feasible solutions from requester i to task executor j. Note that $\mathcal{L}_{i,j}$ has a potentially exponential number of feasible solutions with respect to the decision variables $\boldsymbol{G}$, $\boldsymbol{O}$, $\boldsymbol{A}$, and $\boldsymbol{D}$.

Then, the allocation problem can be formulated from the original (MSW) as

$$\max_{\hat{\boldsymbol{X}}} \sum_{i\in\mathcal{U}} \sum_{j\in\mathcal{N}_i} \sum_{\ell\in\mathcal{L}_{i,j}} w^\ell_{i,j} x^\ell_{i,j} \tag{EQMSW}$$

$$\text{s.t. } C_{11} \sum_{j\in\mathcal{N}_i} \sum_{\ell\in\mathcal{L}_{i,j}} x^\ell_{i,j} \le 1, \; \forall\, i \in \mathcal{U};$$

$$C_{12}: \sum_{i\in\mathcal{U}} \sum_{\ell: t\in l^2_{i,j}(\ell)\in l^2_{i,j}} r^k_{i,j} x^\ell_{i,j} \le R^k_j, \forall\, t \in \mathcal{T}, \forall\, j \in \mathcal{M}, \; k \in \mathcal{K};$$

$$C_{13}: \sum_{i\in\mathcal{U}} \sum_{j\in\mathcal{N}_i} \sum_{\ell: t\in l^1_{i,j}(\ell)\in l^1_{i,j}} \phi_{i,j} x^\ell_{i,j} \le W, \quad \forall\, t \in \mathcal{T};$$

$$C_{14}: x^\ell_{i,j} \in \{0, 1\}, \; \forall\, i \in \mathcal{U}, \; j \in \mathcal{N}_i, \; \ell \in \mathcal{L}_{i,j},$$

where $\hat{\boldsymbol{X}} = \{x^\ell_{i,j}, i \in \mathcal{U}, \, j \in \mathcal{N}_i, \, \ell \in \mathcal{L}_{i,j}\}$ are new decision variables; $w^\ell_{i,j} = v_{i,j} - c_j e^\ell_{i,j}$, where $e^\ell_{i,j}$ is the energy consumption at task executor j when the ℓ-th feasible scheduling scheme is selected. In order to devise an online mechanism with sound CR, we resort to its dual problem. The dual problem of (EQMSW) can be formulated as follows by relaxing the constraint C_{14} into any value between 0 and 1.

$$\min_{u, \hat{p}} \sum_{i\in\mathcal{U}} u_i + \sum_{t\in\mathcal{T}} W \hat{p}_t + \sum_{t\in\mathcal{T}} \sum_{j\in\mathcal{M}} \sum_{k\in\mathcal{K}} R^k_j \hat{p}^k_{j,t} \tag{EQDP}$$

$$\text{s.t. } C_{15}: u_i + \sum_{t\in l^2_{i,j}(\ell)} \sum_{k\in\mathcal{K}} r^k_{i,j} \hat{p}^k_{j,t} + \sum_{t\in l^1_{i,j}(\ell)} \phi_{i,j} \hat{p}_t \ge w_{i,j}, \forall\, i \in \mathcal{U}, \, j \in \mathcal{N}_i, \, \ell \in \mathcal{L}_{i,j};$$

$$C_{16}: u_i \ge 0, \; \hat{p}^k_{j,t} \ge 0, \; \hat{p}_t \ge 0, \; \forall\, i \in \mathcal{U}, \; j \in \mathcal{M}_i, \; k \in \mathcal{K},$$

where u_i, $\hat{p}_t$ and $\hat{p}^k_{j,t}$ are the dual variables corresponding to the constraints C_{11}, C_{12}, and C_{13}, respectively. Note that the dual variables $\hat{p}^k_{j,t}$ can be interpreted as the marginal price of task executor j's available computation frequencies and storages resources (i.e., $k = 1$ or 2) at time slot t, while dual variable $\hat{p}_t$ can be regarded as the marginal price of the available bandwidth in the network. Thus, $\sum\limits_{t \in l^2_{i,j}(\ell)} \sum\limits_{k \in \mathcal{K}} r^k_{i,j} \hat{p}^k_{j,t}$ and $\sum\limits_{t \in l^1_{i,j}(\ell)} \phi_{i,j} \hat{p}_t$ represent the total computation cost and the total transmission cost, respectively. Moreover, u_i can be considered as the utility of requester i. In the following sections, we will apply these observations to design an online mechanism to address problem (MSW).

3.2.2 Online Mechanism

In our formulated online mechanism, we need to decide whether to accept a new task upon its arrival and which task executor should be assigned as well as how much the requester should be charged. Our basic idea is that if the BS decides to assign the current requester i's task to task executor j, we increase the unit price of task executor j's resource based on the fact that it will have less resources, and then apply these updated prices to decide the acceptance of future arrived requesters.

1) *Allocation Rule*: Under the consideration of IC and IR, u_i in constraint C_{13} has to be maximized and greater than zero. In addition, according to the Karush-Kuhn-Tucker (KKT) condition [86] in the prima-dual framework, if requester i is accepted (i.e., $x^\ell_{i,j} = 1$), we have

$$u_i = w_{i,j} - \Big(\sum_{t \in l^2_{i,j}(\ell)} \sum_{k \in \mathcal{K}} r^k_{i,j} \hat{p}^k_{j,t} + \sum_{t \in l^1_{i,j}(\ell)} \phi_{i,j} \hat{p}_t \Big). \tag{23}$$

Combining these two requirements together, u_i can be written as

$$u_i = \max\{0, \max_{\substack{j \in \mathcal{N}_i, \\ \ell \in \mathcal{L}_{i,j}}} \{ w_{i,j} - \Big(\sum_{t \in l^2_{i,j}(\ell)} \sum_{k \in \mathcal{K}} r^k_{i,j} \hat{p}^k_{j,t} + \sum_{t \in l^1_{i,j}(\ell)} \phi_{i,j} \hat{p}_t \Big)\}\}. \tag{24}$$

From (24), we can design the following allocation rule. Upon the arrival of requester i, we choose a task executor in the set $\mathcal{N}_i$ and a scheduling scheme in set $l_{i,j}$ so that u_i is maximized. We denote such best task executor and the scheduling scheme as j^* and $l_{i,j}(\ell^*)$, respectively. Note that the scheduling scheme $l_{i,j}(\ell^*)$, which maximizes the utility of requester i, is referred to as the optimal scheduling scheme at the collaborator j. If at optimum, (23) is larger than zero, requester i's task is accepted; otherwise, it is rejected. Note that we also refer to the above allocation rule as the acceptance condition in this chapter.

2) *Payment Design*: As indicated before, the marginal prices increase with the acceptance of requesters and the designed updating rule is vital to the achievable competitive ratio of our online auction which will be discussed later. The designed marginal price updating rule should follow the following three requirements: (i) at the beginning of the auction, the price should be set sufficiently low in order to allow the acceptance of coming requesters; (ii) after allocating resources for each accepted requester, prices should be increased rapidly to save resources for the future requesters with high valuations; and (iii) if some resources of any task executor are run out at certain time slot, the prices should be set high enough so that no requesters' tasks can be accepted. By considering all these requirements, for any task executor j, we design the marginal prices updating rule as follows

$$\hat{p}_{j,t}^k = \hat{p}_{j,t}^k(1 + \frac{r_{i,j}^k}{R_j^k}) + \frac{r_{i,j}^k}{\Gamma_{i,j} R_j^k},\ \forall\, t \in [g_{i,j}, o_{i,j}],\ \forall\, k \in \mathcal{K}, \tag{25}$$

$$\hat{p}_t = \hat{p}_t(1 + \frac{\phi_{i,j}}{W}) + \frac{\phi_{i,j}}{\Phi_{i,j} W},\ \forall\, t \in [a_{i,j},\, d_{i,j}], \tag{26}$$

where $\Gamma_{i,j} = \frac{\sum_{k\in\mathcal{K}} r_{i,j}^k I_{i,j}^C}{w_{min}}$, $\Phi_{i,j} = \frac{I_{i,j}^T \phi_{i,j}}{w_{min}}$, and w_{min} is the minimal valuable of $w_{i,j}$, which can be estimated from the historical data, and both $\Gamma_{i,j}$ and $\Phi_{i,j}$ can be calculated based on the outputs of the allocation rule. Thus, the price for a requester i to pay can be determined by

$$\begin{cases} p_{i,j} = p_{i,j}^1 + p_{i,j}^2 = \sum\limits_{t\in l_{i,j}^2(\ell)} \sum\limits_{k\in\mathcal{K}} r_{i,j}^k \hat{p}_{j,t}^k + \sum\limits_{t\in l_{i,j}^1(\ell)} \phi_{i,j}\hat{p}_t + e_{i,j}c_j, & \\ & \text{if } i \text{ is accepted;} \\ p_{i,j} = 0, & \text{if } i \text{ is rejected;} \end{cases}$$

3) *Scheduling Design*: To implement Algorithm 6, the maximization problem in (24) needs to be solved. Since we may confront exponential numbers of feasible solutions, it is inefficient to find

the best solution through exclusive searching. To address this issue, we propose a new polynomial time method as follows.

From (24), the original optimization problem can be equivalently converted to one which minimizes the summation of $p^1_{i,j}$, $p^2_{i,j}$, and $e_{i,j}c_j$. Note that since we try to arrange a certain number of time slots to complete the transmission and computation processes for a task, the newly formulated problem for requester i offloading task to the task executor j becomes

$$\begin{aligned} \beta_{i,j} = &\min_{\substack{y_j(t),z_j(t),\\ N_{\phi_j},N_{f_{i,j}}}} \sum_{t\in[t_i,t_i+\tau_i]} \{\frac{h(t)}{N_{\phi_j}} y_j(t) + (\frac{c_1(t)}{N_{f_{i,j}}} + c_2(t))z_j(t)\} + \frac{c_3}{N^2_{f_{i,j}}} \\ \text{s.t.}\ \ & C_{15}: \ y_j(t) < z_j(t), \ \forall\, t \in [t_i, t_i+\tau_i]; \\ & C_{16}: \sum_{t\in[t_i,t_i+\tau_i]} y_j(t) = N_{\phi_j}; \\ & C_{17}: \sum_{t\in[t_i,t_i+\tau_i]} z_j(t) = N_{f_{i,j}}; \\ & C_{18}: \ y_j(t) \in \{0,1\}, \ z_j(t) \in \{0,1\}, \ t \in [t_i, t_i+\tau_i]. \end{aligned} \tag{TSP}$$

where $h(t) = \frac{s_i \hat{p}_t}{\Delta t \log 2(1+\gamma_{i,j})}$, $c_1(t) = \frac{Q_i \hat{p}^1_{j,t}}{\Delta t}$, $c_2(t) = s_i \hat{p}^2_{j,t}$, and $c_3 = \frac{c_j Q_i^3 \xi_j}{\Delta t^2}$ for any pair i and j; $y_j(t)$ and $z_j(t)$ are two new binary scheduling decision variables. If $y_j(t)$ or $z_j(t)$ equals 1, it means requester i transmits the task to task executor j or task executor j executes the task at time slot t, respectively; N_{ϕ_j} and $N_{f_{i,j}}$ denote the total required transmission and computation time slots at task executor j, respectively. Due to the integral decision variables and the nonlinear objective, it's nontrivial to solve problem (TSP) directly. Instead, we decouple it by letting the optimal dividing time slot between transmission period and computation period be $\bar{t}_{i,j} \in [t_i, t_i+\tau_i]$. Then, the scheduling problem (TSP) can be equivalently transformed into two subproblems as

$$\begin{aligned} \beta^1_{i,j} = &\min_{y_j(t),N_{\phi_j}} \sum_{t\in[t_i,\bar{t}_{i,j}]} \frac{h(t)}{N_{\phi_j}} y_j(t) \\ \text{s.t.}\ \ & C_{16}, \text{ and } y_j(t) \in \{0,1\} \end{aligned} \tag{SubP1}$$

$$\beta_{i,j}^2 = \min_{z_j(t), N_{f_{i,j}}} \sum_{t \in (\bar{t}_{i,j}, t_i+\tau_i]} (\frac{c_1(t)}{N_{f_{i,j}}} + c_2(t)) z_j(t) + \frac{c_3}{N_{f_{i,j}}^2} \qquad \text{SubP2}$$

$$\text{s.t. } C_{17}, \text{ and } z_j(t) \in \{0,1\}$$

Lemma 5. *The optimal solution of subproblem* (SubP1) *is obtained when* $N_{\phi_j} = 1$, $t^* = \arg\min_{t \in [t_i, \bar{t}_{i,j}]} h(t)$.

Proof. The proof of contradiction method is applied to prove our statement. We first sort $h(t)$ during the period of $[\bar{t}_{i,j}, t_i + \tau_i]$ in a non-decreasing order into $h(t^1) \le h(t^2) \le h(t^3) \le \cdots$. According to **Lemma** 5, we choose $h(t^1)$ as the optimal solution of (SubP1). On the other hand, if there exist $N_{\phi_j} = N$ continuous transmission time slots, for example $h(t^{n_1}), h(t^{n_2}), \cdots, h(t^{n_N})$, whose $\beta_{i,j}^1$ is smaller than $h(t^1)$, then, we have

$$\frac{h(t^{n_1}) + h(t^{n_2}) + \cdots h(t^{n_N})}{N} < h(t^1). \tag{27}$$

However, this contradicts with the fact that $h(t^1) \le h(t^{n_v}), v = 1, 2, \cdots, N$. Thus, our conclusion holds for (SubP1). This completes the proof. ■

Lemma 6. *Let* $\beta_{i,j}^{2,1}(N) = \sum_{t \in [1, +\infty]} (\frac{c_1(t)}{N} + c_2(t)) z_j(t)$ *be the value under the optimal scheduling when* $N_{f_{i,j}} = N$ *and let* $\beta_{i,j}^{2,2}(N) = \frac{c_3}{N^2}$. *Then, we have* $\beta_{i,j}^{2,1}(N)$ *is an increasing function with respective to* N *and there exists at most one intersection point between* $\beta_{i,j}^{2,1}(N)$ *and* $\beta_{i,j}^{2,2}(N)$.

Proof. This statement is obtained by using the analytical approach. We first compare objective values of $\beta_{i,j}^{2,1}(N)$ and $\beta_{i,j}^{2,1}(N+1)$. Let $t^{n_1}, t^{n_2}, \cdots t^{n_N}$ be the best N numbers of continuous time slots, which means when $N_{f_{i,j}} = N$, the objective of (SubP1) is minimized by selecting these time slots. Likewise, denote $t^{m_1}, t^{m_2}, \cdots t^{m_{(N+1)}}$ as the optimal continuous time slots when $N_{f_{i,j}} = N + 1$. Then, we have

$$\beta_{i,j}^{2,1}(N) - \beta_{i,j}^{2,1}(N+1) = \frac{\sum_{v=1}^{N} c_1(t^{n_v})}{N} + \sum_{v=1}^{N} c_2(t^{n_v}) - (\frac{\sum_{v=1}^{N+1} c_1(t^{m_v})}{N+1} + \sum_{v=1}^{N+1} c_2(t^{m_v}))$$

$$\Rightarrow (N+1)(\beta_{i,j}^{2,1}(N) - \beta_{i,j}^{2,1}(N+1))$$

$$= (N+1)(\frac{\sum_{v=1}^{N} c_1(t^{n_v})}{N} + \sum_{v=1}^{N} c_2(t^{n_v})) - (\sum_{v=1}^{N+1} c_1(t^{m_v}) + (N+1)\sum_{v=1}^{N+1} c_2(t^{m_v}))$$

$$= (N+1)\underbrace{(\frac{\sum_{v=1}^{N} c_1(t^{n_v})}{N} + \sum_{v=1}^{N} c_2(t^{n_v}))}_{A} - N\underbrace{(\frac{\sum_{v=1}^{N} c_1(t^{m_v})}{N} + \sum_{v=1}^{N} c_2(t^{m_v}))}_{B} -$$

$$\underbrace{(\frac{Nc_1(t^{m_{N+1}})}{N} + \sum_{v=1}^{N} c_2(t^{m_v}))}_{C} - (N+1)c_2(t^{m_{N+1}}). \tag{28}$$

Since A is the optimal objective value when $N_{f_{i,j}} = N$, we have $(N+1)\times A < N\times B+C$. Thus, we have $\beta_{i,j}^{2,1}(N) < \beta_{i,j}^{2,1}(N+1)$, which means $\beta_{i,j}^{2,1}(N)$ is an increasing function with respective to N. Moreover, $\beta_{i,j}^{2,2}(N)$ is a decreasing function with respective to N and $\beta_{i,j}^{2,2}(+\infty) = 0 < \beta_{i,j}^{2,1}(+\infty)$. Thus, $\beta_{i,j}^{2,1}(N)$ and $\beta_{i,j}^{2,2}(N)$ have one intersection point only when $\beta_{i,j}^{2,1}(N) = \beta_{i,j}^{2,2}(N)$. This completes the proof. ■

Based on **Lemma** 5, the allocated transmission bandwidth for requester i is always $\phi_{i,j} = \frac{s_i}{\Delta t \log_2(1+\gamma_{i,j})}$. According to **Lemma** 6, there must exist a $\overline{N}$ which can minimize the value of $\beta_{i,j}^{2,1}(N) + \beta_{i,j}^{2,2}(N)$. Note that $\beta_{i,j}^{2,1}(N) + \beta_{i,j}^{2,2}(N)$ decreases when $N < \overline{N}$, but increases when $N > \overline{N}$. If there are M_2 available time slots during $[\bar{t}_{i,j}, t_i + \tau_i]$, we apply the following strategies to get the optimal solution of subproblem (SubP2).

- If $\beta_{i,j}^{2}(1) > \beta_{i,j}^{2}(M_2-1) > \beta_{i,j}^{2}(M_2)$, we choose $\beta_{i,j}^{2}(M_2)$ and the corresponding scheduling scheme, denoted as the set π^*, as the optimal solution, as shown in Fig. 3.2;
- Otherwise, we apply sequential search to compare the values of $\beta_{i,j}^{2}(N+1)$ and $\beta_{i,j}^{2}(N)$ till $\beta_{i,j}^{2}(N+1) > \beta_{i,j}^{2}(N)$. We then choose $\beta_{i,j}^{2}(N)$ and the corresponding scheduling scheme, denoted as the set π^*, as the optimal solution, as shown in Fig. 3.3.

Obviously, for the worst case, we only need $\frac{(\overline{N}+1)(2M_2-\overline{N})}{2} + \overline{N}$ comparisons to reach the optimal solution, which is much more computationally efficient compared to the brute force approach. The detailed procedures for solving the scheduling problem are summarized in **Algorithm** 5. Obviously, **Algorithm** 5 can find the globally optimal solution for the scheduling problem (TSP).

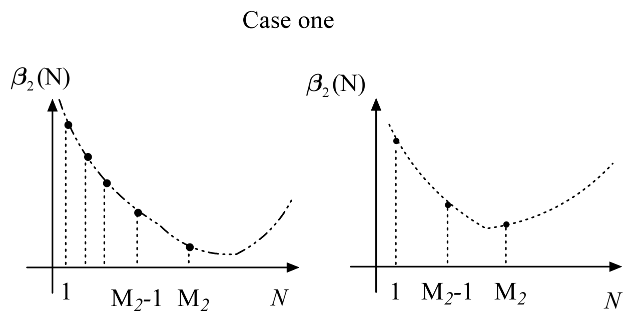

Figure 3.2: Relationship between $\beta_2(N)$ and N.

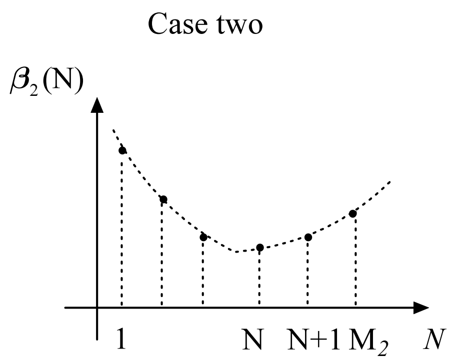

Figure 3.3: Relationship between $\beta_2(N)$ and N.

We summarize the proposed online mechanism integrating allocation rule and payment design in **Algorithm** 6.

3.2.3 Performance Analyses

In this section, we will theoretically analyze our proposed online mechanism in terms of competitive ratio, feasibility of primal and dual solutions, CE, IC, and IR.

Lemma 7. *The competitive ratio of our proposed online mechanism is 3.*

Proof. Assume that the requester i offloads its task to task executor j, and we define $\Delta P^{(i)}$ and $\Delta D^{(i)}$ as the increment of objective values in primal and its dual problems after requester i has

Algorithm 5: Online Auction for Scheduling Problem.

Input: s_i, Δt, $\hat{p}_t$, $\hat{p}_{j,t}^k$, t_i, τ_i, and $\gamma_{i,j}$

Output: Optimal schedule $l_{i,j}(\ell)$ and minimum $\beta_{i,j}$ for requester i offloading task to requester j or BS.

Initialization;

$l_{i,j}^1(\ell) = \emptyset, l_{i,j}^2(\ell) = \emptyset$, and $\beta_{i,j} = +\infty$;

while $\bar{t}_{i,j} \in [t_i, t_i + \tau_i]$ **do**

  $t^* = \underset{t \in [t_i, \bar{t}_{i,j}]}{\arg\min}\, h(t)$ and get $\beta_{i,j}^1$ ▷ Solve (SubP1);

  Apply the above strategies for (SubP2) in $[\bar{t}_{i,j}, t_i + \tau_i]$ and get $\beta_{i,j}^2$ as well as π^*;

  if $\beta_{i,j} > \beta_{i,j}^1 + \beta_{i,j}^2$ **then**

    $\beta_{i,j} = \beta_{i,j}^1 + \beta_{i,j}^2$;

    $l_{i,j}^1(\ell) = l_{i,j}^1(\ell) \cup t^*$ and $l_{i,j}^2(\ell) = l_{i,j}^2(\ell) \cup \pi^*$;

    $l_{i,j}(\ell) = l_{i,j}^1(\ell) \cup l_{i,j}^2(\ell)$;

  Move $\bar{t}_{i,j}$ to the next time slot in $[t_i, t_i + \tau_i]$;

return $l_{i,j}(\ell)$ and $\beta_{i,j}$;

Algorithm 6: Online Auction for Collaborative Task Offloading in MEC.

Input: $w_{i,j}$, s_i, Δt, $\hat{p}_t$, $\hat{p}_{j,t}^k$, t_i, and τ_i

Output: Optimal schedule $l_{i,j}(\ell^*)$, j^* and payment $p_{i,j}$.

Initialization;

$x_{i,j}^\ell = 0,\ \forall i \in \mathcal{U},\ \forall j \in \mathcal{N}_i,\ \ell \in \mathcal{L}_{i,j}$;

$u_i = 0$; ▷the utility of requester i;

$j^* = \emptyset$ and $l_{i,j}(\ell^*) = \emptyset$;

while *the arrival of requester i's task* **do**

  for $j \in \mathcal{N}_i$ **do**

    Run **Algorithm** 5 to get the best scheduling scheme $l_{i,j}(\ell)$ and minimum $\beta_{i,j}$;

    if $w_{i,j} - \beta_{i,j} > u_i$ **then**

      $u_i = w_{i,j} - \beta_{i,j}$;

      $j^* = j$;

      $l_{i,j}(\ell^*) = l_{i,j}(\ell)$;

  if $u_i > 0$ **then**

    Accept requester i and set $x_{i,j^*}^{\ell^*} = 1$;

    Allocate the collaborator or BS and implement schedule scheme according to j^* and $l_{i,j}(\ell^*)$;

    Charge requester i at price $p_{i,j}$;

    Update $\hat{p}_{j,t}^k$ and $\hat{p}_t$ based on (25) and (26);

  else

    Reject requester i and set $x_{i,j}^\ell = 0$ and $p_{i,j} = 0$;

been served, respectively. Then, we have

$$
\begin{aligned}
\Delta D^{(i)} &= u_i + \sum_{t\in\mathcal{T}} W\Delta\hat{p}_t + \sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}} R_j^k \Delta\hat{p}_{j,t}^k \\
&= w_{i,j} - \sum_{a_{i,j}\le t\le d_{i,j}} \phi_{i,j}\hat{p}_t - \sum_{a_{i,j}\le t\le d_{i,j}}\sum_{k\in\mathcal{K}} r_{i,j}^k \hat{p}_{j,t}^k + \sum_{t\in\mathcal{T}} W\Delta\hat{p}_t + \sum_{t\in\mathcal{T}}\sum_{k\in\mathcal{K}} R_j^k \Delta\hat{p}_{j,t}^k \\
&= w_{i,j} - \sum_{a_{i,j}\le t\le d_{i,j}} \phi_{i,j}\hat{p}_t - \sum_{a_{i,j}\le t\le d_{i,j}}\sum_{k\in\mathcal{K}} r_{i,j}^k \hat{p}_{j,t}^k + \sum_{a_{i,j}\le t\le d_{i,j}} (\phi_{i,j}\hat{p}_t + \frac{\phi_{i,j}}{\Phi_{i,j}}) \\
&\qquad + \sum_{a_{i,j}\le t\le d_{i,j}}\sum_{k\in\mathcal{K}} (r_{i,j}^k \hat{p}_{j,t}^k + \frac{r_{i,j}^k}{\Gamma_{i,j}}) \\
&= w_{i,j} + \sum_{a_{i,j}\le t\le d_{i,j}} \frac{\phi_{i,j}}{\Phi_{i,j}} + \sum_{a_{i,j}\le t\le d_{i,j}}\sum_{k\in\mathcal{K}} \frac{r_{i,j}^k}{\Gamma_{i,j}} \le 3w_{i,j} = 3\Delta P^{(i)}.
\end{aligned}
$$

Let $\mathcal{U}^*$ be the set of the offloaded requesters, and $\overline{P}$ and $\overline{D}$ be solutions of primal and its dual problems by our online mechanism, respectively. Then, we must have

$$\overline{P} = \sum_{i\in\mathcal{U}^*} \Delta P^{(i)} = 3\sum_{i\in\mathcal{U}^*} \Delta D^{(i)} = 3\overline{D}.$$

From the linear dual theory, we have

$$\frac{P^*}{\overline{P}} \le \frac{\overline{D}}{\overline{P}} = 3,$$

where P^* is the optimal solution of primal problem (EQMSW). This completes the proof. ■

Lemma 8. *Our proposed online mechanism produces almost feasible solutions to offline problem* (EQMSW) *if* $W \gg 1$*;* $R_j^k \gg 1, \forall j$*;* $r_{i,j}^k \ll R_j^k$*, and* $\phi_{i,j} \ll W, \forall i, j$*.*

Proof. Please refer to Appendix B.1. ■

Lemma 9. *Our proposed online mechanism produces a feasible solution to dual problem* (EQDP).

Proof. We consider the following two cases.

- Case 1: Requester i is rejected, which means $w_{i,j^*} - \beta_{i,j^*} \le 0$ for the best selected task executor j^* and $u_i = 0$ according to the acceptance condition (24). Thus, constraint C_{15} holds in this case.

- Case 2: Requester i is accepted, which means $u_i = w_{i,j^*} - \beta_{i,j^*} > 0$ for the best selected task executor j^*. Thus, constraint C_{15} still holds in this case.

Therefore, constraint C_{15} in problem (EQDP) always holds no matter whether requester i is accepted or not. This completes the proof. ■

Lemma 10. *Our proposed online mechanism runs in polynomial time to get the result.*

Proof. The computational complexity of the proposed online mechanism is evaluated in terms of computation times with respect to the number of requesters and collaborators. Recall that our proposed online mechanism consists of **Algorithm** 6 and **Algorithm** 5. For **Algorithm** 5, given that there are total M time slots during the period of $[t_i, t_i + \tau_i]$, the computational complexity of **Algorithm** 5 can be calculated as $O(M(M - M + 1 + \frac{M(M-1)}{2} + M - 1) = O(M^2 \frac{(M+1)}{2})$. Therefore, the computational complexity of **Algorithm** 6 is $O(|\mathcal{U}| \times |\mathcal{N}_{max}| \times M^2 \frac{(M+1)}{2})$. Note that this is the worst case computational complexity. Obviously, **Algorithm** 6 runs in polynomial time, which completes the proof. ■

Lemma 11. *The proposed online mechanism can guarantee IC and IR.*

Proof. Please refer to Appendix B.2. ■

Theorem 5. *The proposed online mechanism has a competitive ratio of 3, runs polynomially, and guarantees truthfulness and individual rationality.*

Proof. By combing **Lemma 7**, **Lemma 10**, and **Lemma 11**, we can get the above conclusion. This completes the proof. ■

3.3 Numerical Simulations

In this section, numerical simulations are conducted to verify the effectiveness of our proposed online mechanism. Since the total social welfare, revenue, and utility of requester are the most important economical metrics and the competitive ratio is also a vital metric to measure an online mechanism, in this section, we will focus on evaluating these two performance metrics with respect

Table 3.2: Main Simulation Parameters In Chapter 3

Parameter	Value
Cell radius	500 m
Total bandwidth	40 MHz
Transmission power at requesters	1.5 W
Background noise average power	-60 dBm
Total running time	30 minutes
Time slot length	1 second
Task size	Randomly from 10 to 30 MB
CPU cycles coefficient	330 cycles/Byte
Energy consumption coefficient	10^{-26}
Unit energy cost	$\$0.1$
Valuation	Randomly from $[\$0.1, \$10]$
The maximum delay	Randomly from [5, 15] seconds
Computation capacity of BS	10 GHz
Storage capacity of BS	10 GB
Computation capacity of collaborators	2 GHz
Storage capacity of collaborators	5 GB

to different numbers of requesters and collaborators. In the simulation, the wireless channels between requesters and task executors (i.e., collaborators or the BS) experience Rayleigh fading and all the channel coefficients are zero-mean, Circularly Symmetric Complex Gaussian (CSCG) random variables with variances d^{-v}, where d is the distance between the transmitter and the receiver and $v = 4$. Table 3.2 lists the main simulation parameters, some of which have also been employed in [71, 101, 102]. For comparison purpose, the following three online strategies are also simulated as benchmarks.

- Random online mechanism: For each requester, the BS randomly selects the task executor and randomly schedules the transmission and computation times.
- Greedy online mechanism: Upon the arrival of a requester, the BS chooses the task executor with the maximal valuation as the winner and schedules one time slot for transmission and $\lceil \tau \rceil - 1$ time slots for computation.
- First In First Out (FIFO) online mechanism [103]: Arriving tasks are always accepted with a fixed transmission and computation time schedule till the resources are run out.

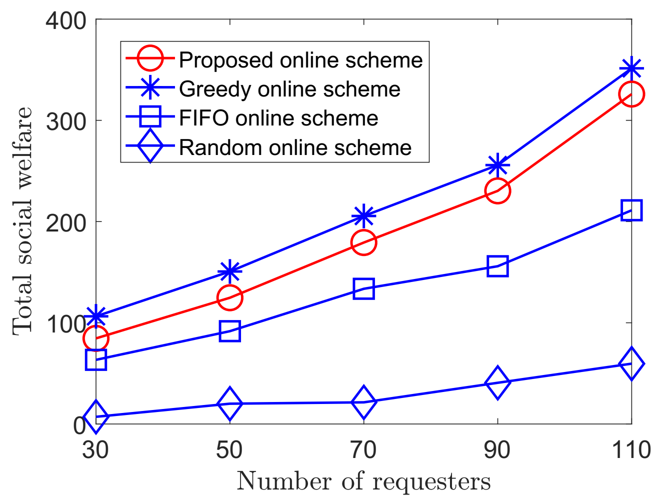

Figure 3.4: TSW versus numbers of requesters.

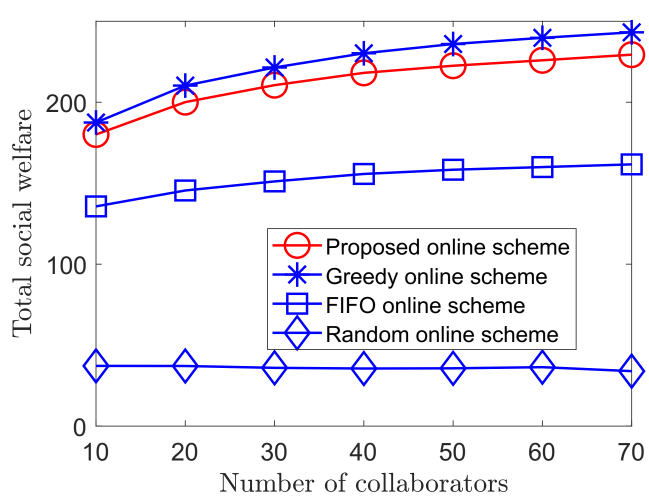

Figure 3.5: TSW versus numbers of collaborators.

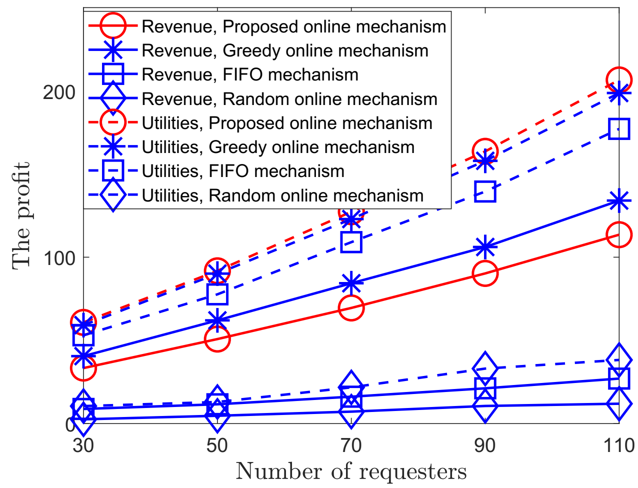

Figure 3.6: Profits versus numbers of requesters.

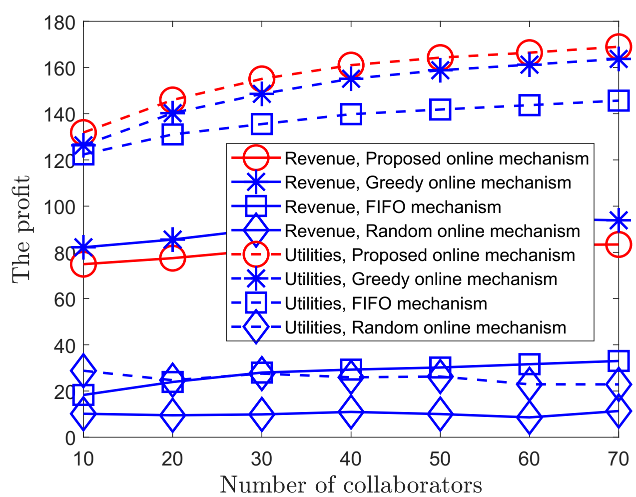

Figure 3.7: Profits versus numbers of collaborators.

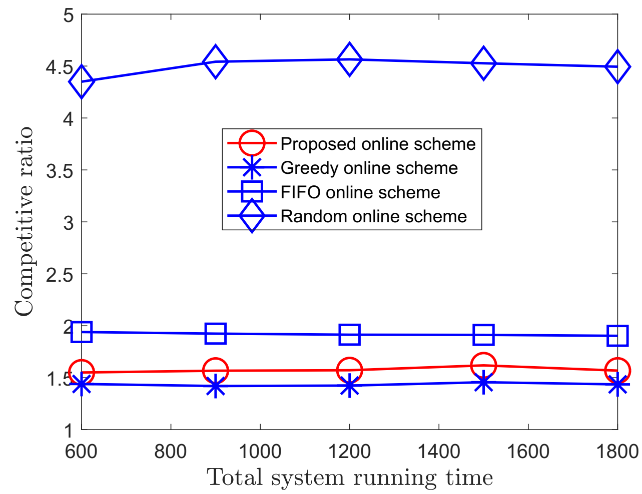

Figure 3.8: CR versus total system running time.

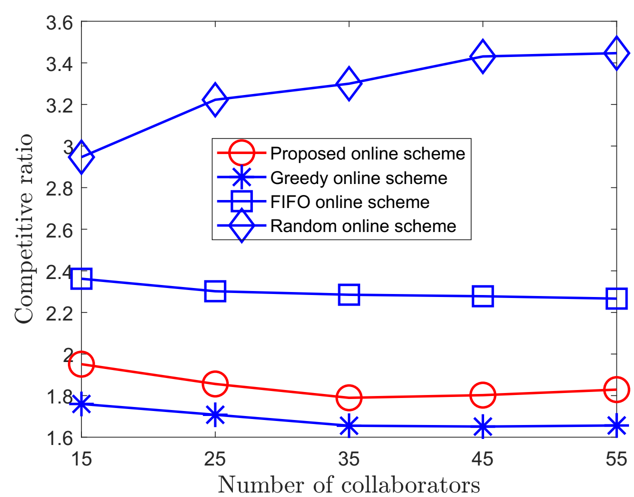

Figure 3.9: CR versus numbers of collaborators.

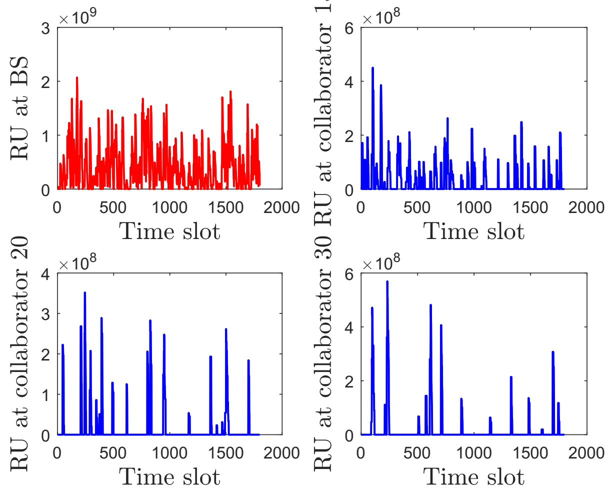

Figure 3.10: Utilized computation resource by proposed mechanism.

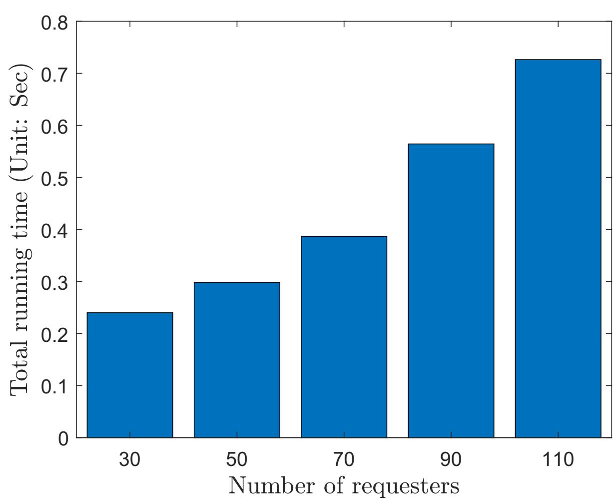

Figure 3.11: Time consumption of the whole system

Fig. 3.4 shows the Total Social Welfare (TSW) achieved by different online mechanisms with respect to different numbers of arrived requesters when there are 30 collaborators, i.e., $|\mathcal{M}| = 30$. From this figure, we can see that the achievable total social welfare increases with the number of requesters. This trend is obvious since with the arrival of more requesters, the BS will admit more before resources are exhausted which results in the increment on the total social welfare. It is worthy noting that our proposed online mechanism outperforms both the random and FIFO online mechanisms, but underperforms the greedy one. This is because the proposed online mechanism tries to minimize the scheduling problem (TSP) so as to maximize the utility of each requester, while the greedy one only attempts to maximize the total social welfare and ignores the maximality of the individual utility. In addition, according to [104], the simple greedy online mechanism cannot guarantee the truthfulness and individual rationality properties.

Fig. 3.5 reevaluates the total social welfare under various numbers of collaborators. In the simulation, the total number of arrived requesters is fixed at 75, i.e., $|\mathcal{U}| = 75$. It can be seen from the figure that the total social welfare increases with the number of collaborators till reaching a saturation when the number of collaborators is large enough (e.g., 55 in our simulation). This is because with the excessive amount of collaborators, each requester is always served by its most effective collaborator while other collaborators have no effects on the achievable total social welfare. In addition, the total social welfare of random online mechanism is almost a constant. The reason is that this mechanism selects the collaborator in random so that it treats all collaborators equally regardless of how many collaborators exist. Similar to Fig. 3.4, our proposed online mechanism outperforms both the random and FIFO online mechanisms, but is still inferior to the greedy one. In summary, from both Figs. 3.4 and 3.5, we can conclude that although it is not difficult to design an online algorithm with a sound competitive ratio (or larger social welfare), it does be hard to devise an online mechanism which possesses sound competitive ratio, truthfulness, and individual rationality at the same time. In fact, our proposed online mechanism scarifies a little bit of competitive ratio to achieve other economical properties.

Fig. 3.6 shows relationship between revenues or utilities and the number of requesters under different online mechanisms. All other simulation parameters are set the same as those in Fig. 3.4. Revenues are calculated by summing the substraction of each task executor's payment and its cost while utilities are the summation of all requesters. From this figure, we can observe that both revenues and utilities increase with the number of requesters. This is because more requesters can be accepted, which can gain more benefits. Note that since the payment by the greedy online mechanism is larger than the proposed one, the revenue of greedy online mechanism must be higher than that of the proposed one. But, utilities of the proposed online mechanism are larger than all other mechanisms. Moreover, both the differences of revenues and utilities between proposed and greedy online mechanisms are gradually increasing when the number of requesters increases. The reason behind this can be explained as follows. When the number of requesters are small, there are no obvious difference between the proposed and greedy online mechanisms in terms of scheduling scheme due to the fact that since the number of requesters are small enough compared to the system total time slots, the marginal prices at each time slot are small enough and have little effects on total payment. However, such effects increase with the increase numbers of requesters.

Fig. 3.7 illustrates profits of different mechanisms in terms of both revenues and utilities with respect to the number of collaborators when $|\mathcal{U}| = 75$. It can be observed that profits of the proposed, greedy, and FIFO mechanisms increase first and then keep stable when the number of collaborators becomes large enough, while the profit of the random online mechanism is a constant. The reasons are the same as those to explain the trend in Fig. 3.5. Moreover, even though the revenue of greedy online mechanism is larger than that of the proposed one, the utilities of requesters are lower than the proposed online mechanism. This further verifies the effectiveness of our proposed online mechanism and the conclusions we have drawn from Fig. 3.4.

Fig. 3.8 presents the comparison among different online mechanisms in terms of the competitive ratio by varying the system running time when the number of collaborators is 25. Note that the optimal offline solution is obtained by *Yalmip* optimizer and the payments are based on the VCG mechanism [3]. From Fig. 3.8, we can observe that the CR of our proposed online mechanism is less than 3, which matches our theoretical analyses. Besides, the competitive ratio almost stays unchange with different system running times, which demonstrates that the proposed online truthful mechanism is stable.

Fig. 3.9 evaluates the performance of different online mechanisms in terms of competitive ratio with respective to different numbers of collaborators when the number of requesters is 40. For the proposed, FIFO, and greedy online mechanisms, their competitive ratios keep less than 3 and slightly decrease till tending to be stable when the number of collaborators becomes large enough. It is because more collaborators can increase the available resources in the system and increase the number of potential alternative collaborators around requesters. In contrast, the competitive ratio of the random online mechanism increases (i.e., the worse performance) with the number of collaborators. According to Fig. 3.5, as the increase of $|\mathcal{M}|$, the social welfare of random online mechanism stays unchanged, while the offline optimal solution increases because more resources are available for allocation. Thus, the competitive ratio of random online mechanism increases.

Fig. 3.10 depicts the utilization of computation resource of the proposed online mechanism at the BS and some of collaborators with the evolution of time. In our simulation, we set $|\mathcal{N}| = 150$ and $|\mathcal{M}| = 50$. Since there are a lot of collaborators, we randomly choose three collaborators and the BS to observe their computation resource utilizations. As shown in this figure, the maximal computation resources at the BS, collaborators 15, 20, and 30 are around 2.3 GHz, 480 MHz, 380 MHz, and 580 MHz, accordingly. Obviously, those values are less than their provided computation resources, which demonstrates feasibility of our solution.

Fig. 3.11 reveals the time consumption of the whole system with the number of requesters when there are 30 collaborators, i.e., $|\mathcal{M}| = 30$. Obviously, it is intuitive and reasonable that the total running time for the whole system increases with the number of requesters. What's more, we can observe that the execution time for single task is roughly 5 millisecond, and the total running time for all tasks is only a fraction of second, which further demonstrates the computational efficiency of our proposed online incentive mechanism.

3.4 Summary

In this chapter, online truthful incentive mechanism design for collaborative task offloading in EC has been studied. By considering each task's specific requirements in terms of data size, delay, and preference, a social-welfare-maximization problem is formulated. After that, an effective online mechanism is developed based on the primal-dual framework to properly select task executors, suitably schedule transmission and computation times, and optimally allocate the transmission and computation resources. Both theoretical and numerical results show that our proposed mechanism can guarantee feasibility, truthfulness, and computational efficiency with competitive ratio of 3.

Chapter 4

Nonlinear Online Incentive Mechanism Design in Edge Computing Systems with Energy Budget

In this chapter, we consider task offloading in EC systems, where tasks are offloaded by the base station to resourceful mobile users. With the consideration of unique characteristics in practical edge computing systems, such as dynamic arrival of computation tasks, and energy constraints at battery-powered mobile users, we formulate an incentive mechanism design problem by jointly optimizing task offloading decisions, and allocation of both communications (i.e., power and bandwidth), and computation resources. In order to tackle the nonlinear issue, a newly designed online truthful mechanism is proposed for task offloading. The considered system model consists of IoT devices, a central controller (such as the BS), and multiple mobile users. At the beginning of each time slot, the BS firstly collects requests of offloading tasks from IoT devices and then broadcasts them to mobile users, who will then submit their valuations and available energy to the BS. After that, the BS determines the best mobile user for each task, the transmission power and bandwidth for both upload and download transmissions, the allocated computation resource, and the corresponding payment to mobile users. This process recurs along the time, and decision making in each time slot is correlated because of the energy constraint on each mobile user. With the objective of maximizing social welfare, we first formulate an offline optimization problem and design a nonlinear online truthful mechanism based on the rule of MIDR. Finally, we reconsider energy constraints to design a new nonlinear online incentive mechanism by rationally combining the previously derived one-shot ones. Theoretical analyses show that our proposed nonlinear online incentive mechanism can guarantee individual rationality, truthfulness, a sound competitive ratio, and computational efficiency. We further conduct comprehensive simulations to validate the effectiveness and superiority of our proposed mechanism.

4.1 System model and Problem Formulation

In this section, we first describe the system under consideration, including both computation and communication models, and then formulate the interaction between the BS and mobile users as an online incentive mechanism design problem. After that, the corresponding offline optimization problem is formulated. For the notational convenience, Table 4.1 summarizes the major notations used in this chapter.

4.1.1 Network Artechiture

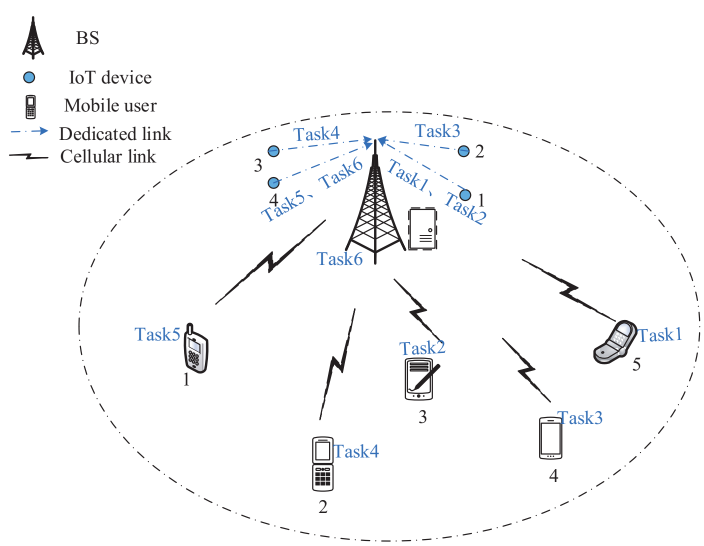

Figure 4.1: The system model with 4 IoT devices and 5 mobile users.

Table 4.1: Commonly Used Notations In Chapter 4

Notation	Interpretation
$\mathcal{T}$	set of time slots
ΔT	time length of each time slot
$\mathcal{N}^t$	Set of task at time slot t
$\mathcal{M}^t$	Set of available mobile users at time slot t
L_j^t	bid of mobile user i at time slot t
c_j^t	submitted unit power cost by mobile user j
v_j^t	true or actual cost of c_j^t
$c_{B,j}^t$	unit power cost at the BS to mobile user j
E_j	total energy of mobile user j
L_i^t	set of submitted bids of the mobile user i
J_i^t	task i at time slot t
S_i^t	input size of task i at time slot t
O_i^t	size of returned result at time slot t
D_i^t	delay tolerance of task i at time slot t
Q_i^t	total required CPU cycles of task i at time slot t
$T_{i,j}^{exe,t}$	total execution time of task i at time slot t
$f_{i,j}^t$	allocated CPU frequency at mobile user j
F_j	total CPU frequency at mobile user j
$C_{i,j}^{exe,t}$	total energy consumption of task i at time slot t
B^U, B^D	total bandwidth for the downlink and uplink in the system
$w_{i,j}^{D,t}$, $w_{i,j}^{U,t}$	allocated downlink and uplink bandwidths for task i
$x_{i,j}^t$	binary decision variable
$e_{i,j}^t$	total energy consumption at mobile user j for task i
$R_{i,j}^{U,t}$, $R_{i,j}^{D,t}$	uplink and downlink transmission rates
$P_{i,j}^{U,t}$, $P_{i,j}^{D,t}$	transmission power at mobile user j and the BS
$T_{i,j}^{U,t}$, $T_{i,j}^{D,t}$	uplink and downlink transmission times for task i
$\pi_{i,j}^t$	reimbursement for mobile user j foe executing task i
$\boldsymbol{L}^t$	collection of all submitted bids at time slot t
$\boldsymbol{L}_j^t$	collection of all submitted bids by mobile user j
$\boldsymbol{L}_{-j}^t$	collection of bids, excluding mobile user j

Similar to [57], consider an edge computing system, as shown in Fig. 4.1. In the system, there are a BS, several IoT devices, and some mobile users who provide computation services in the coverage of the BS. IoT devices generate tasks along the time and submit their tasks at the beginning of each time slot to the BS through dedicated links, such as WiFi or Low-power Wide Area Networking (LPWAN). After receiving all these tasks, the BS determines the task assignment among mobile users and itself, and corresponding computation and communication resource allocations. After that, the BS offloads assigned tasks to the mobile users. Once the completion of tasks, results will be returned to the BS by mobile users. Note that for those unoffloaded tasks, the BS will consume its own computation resource and energy for execution[1]. Define a time-slotted structure with time slots indexed by $\mathcal{T} = \{1, 2, \cdots, T\}$, where T is the total number of time slots, and there is a set of tasks $\mathcal{N}^{(t)}$ at each time slot t with cardinality of $|\mathcal{N}^{(t)}| = N^{(t)}$. At the beginning of each time slot, the BS broadcasts tasks to a set of mobile users $\mathcal{M}^{(t)}$ with cardinality of $|\mathcal{M}^{(t)}| = M^{(t)}$ for execution, and mobile users connect with the BS via the cellular network. Similar to existing work in edge computing [55, 57], a quasi-static scenario is studied[2], where all mobile users and wireless communication configurations keep stationary in each time slot, but may change slot by slot. In addition, following [57, 105], if the task has been successfully offloaded to a mobile user in time slot t, its execution will be completed within the time slot period, i.e., ΔT. Technically, if the size of generated task is too large, the IoT device can split the oversize task into multiple small tasks before offloading [106, 107].

Obviously, the aforementioned interactions could be modelled as an auction, where mobile users are sellers who sell their unused computational resource for monetary benefits, while the BS is the buyer. At the beginning of each time slot t, the BS broadcasts descriptions of $N^{(t)}$ tasks to mobile users, and then the bid from each mobile user j, denoted as $L_j^{(t)} = \{c_j^{(t)}, E_j\}$, is submitted to the BS. Here $c_j^{(t)}$ represents the cost per unit power consumption at mobile user j, and E_j is the total available energy of mobile user j, which is submitted at the first time slot only. We denote the true or actual valuation of $c_j^{(t)}$ by $v_j^{(t)}$, which is private and only known to the mobile user j [51]. After collecting all bids from mobile users, the BS determines which task should be executed by which mobile user and how much reimbursement should be given to winning mobile users.

Following [92, 108], data transmission in the download link (from the BS to mobile users) and upload link (from mobile users to the BS) are mainly considered in this chapter, while the overhead for the control signalling is overlooked. This is because compared to the data size and returned results, control signalling is much smaller, and can be transmitted through dedicated channels.

[1]Since the offloading process between the BS and mobile users is the main focus in this chapter, the computation resource allocation at the BS side is ignored.

[2]Note that, our work can be extended to the mobility case, in which we can assume that mobile users follow a certain known mobility pattern, and introduce the expectation to the objective function. We will study this more complicated mobility case in the future work.

4.1.2 Computation Model

We mathematically characterize each task i at time slot t by a tuple $J_i^{(t)} = \{S_i^{(t)}, O_i^{(t)}, D_i^{(t)}\}$, where $S_i^{(t)}$ denotes the size of the input, $O_i^{(t)}$ denotes the size of the return result, and $D_i^{(t)}$ denotes the delay tolerance. The required amount of CPU cycles $Q_i^{(t)}$ for task i can be estimated as [71]

$$Q_i^{(t)} = \epsilon S_i^{(t)}, \tag{29}$$

where ϵ is the CPU cycle coefficient. Note that the allocated computational frequency at each mobile user j is an optimization variable, which is represented by $f_{i,j}^{(t)}$. Then, the execution time for the allocated task at the mobile user j can be calculated as

$$T_{i,j}^{exe,(t)} = \frac{Q_i^{(t)}}{f_{i,j}^{(t)}},\ \forall\, j \in \mathcal{M}^{(t)},\ \forall\, i \in \mathcal{N}^{(t)}. \tag{30}$$

Furthermore, since the computational capacity of each mobile user j is ordinarily limited, the allocated computation frequency for the task should be no more than this limitation, i.e.,

$$f_{i,j}^{(t)} \leq F_j, \quad \forall\, j \in \mathcal{M}^{(t)}, \tag{31}$$

where F_j denotes the computational capacity of mobile user j. Based on [71] and [99], the energy consumption $E_{i,j}^{exe,(t)}$ (unit: Joule) and energy consumption cost $C_{i,j}^{exe,(t)}$ (unit: dollar) at mobile user j for executing task i can be respectively computed as

$$E_{i,j}^{exe,(t)} = \beta_j Q_i^{(t)} (f_{i,j}^{(t)})^2,\ \forall\, j \in \mathcal{M}^{(t)},\ \forall\, i \in \mathcal{N}^{(t)}, \tag{32}$$

$$C_{i,j}^{exe,(t)} = \theta_j Q_i^{(t)},\ \forall\, j \in \mathcal{M}^{(t)},\ \forall\, i \in \mathcal{N}^{(t)}, \tag{33}$$

where β_j is the energy consumption coefficient at mobile user j, and θ_j is the energy cost per CPU cycle. According to [71, 99], β_j and θ_j can be estimated in practice.

4.1.3 Communication Model

The communication model includes download and upload links, whereby tasks can be offloaded to selected mobile users, and computation results can be fed back to the BS, respectively. Let B^U and B^D be the total bandwidths for the upload and download links, respectively. Define $x_{i,j}^{(t)}$ to be an indicator variable, which means task i is allocated to mobile user j in time slot t if $x_{i,j}^{(t)} = 1$, and $x_{i,j}^{(t)} = 0$ otherwise.

Each mobile user can only execute at most one task at each time slot t, and each task can only be offloaded to at most one mobile user. Those tasks, which are not executed at the current time slot t, will be executed by the BS before their deadlines. For $x_{i,j}^{(t)}$, we have the following constraints

$$\sum_{i \in \mathcal{N}^{(t)}} x_{i,j}^{(t)} \leq 1, \ \forall j \in \mathcal{M}^{(t)}, \forall t \in \mathcal{T}, \tag{34}$$

$$\sum_{j \in \mathcal{M}^{(t)}} x_{i,j}^{(t)} \leq 1, \ \forall i \in \mathcal{N}^{(t)}, \forall t \in \mathcal{T}. \tag{35}$$

In addition, since each mobile user is energy constrained, it has the energy restriction over the time as

$$\sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{N}^{(t)}} e_{i,j}^{(t)} x_{i,j}^{(t)} \leq E_j, \forall j \in \mathcal{M}^{(t)}, \tag{36}$$

where $e_{i,j}^{(t)}$ is the total energy consumption of mobile user i at time slot t. Note that as indicated in the constraint (36), the decision-makings are coupled along the time slots. Moreover, $e_{i,j}^{(t)}$ is a combination of upload link transmission energy and task execution energy, as shown in (40), both of which are related to optimization variables. Hence, by introducing the energy budget on each mobile user, the considered mechanism design problem is far more challenging than traditional ones in the literature.

For the upload link, since mobile users need to transmit results back to the BS, the transmission rate with the allocated bandwidth $w_{i,j}^{U,(t)}$ can be calculated as

$$R_{i,j}^{U,(t)} = w_{i,j}^{U,(t)} \log_2(1 + \gamma_{i,j}^{U,(t)}), \tag{37}$$

where $\gamma_{i,j}^{U,(t)} = \frac{P_{i,j}^{U,(t)} G_{i,j}^{U,(t)}}{\sigma^2}$ is the SNR. $P_{i,j}^{U,(t)}$ and $G_{i,j}^{U,(t)}$ are the transmission power and the channel gain between mobile user j and the BS, respectively, and σ^2 represents the background noise power. Note that each mobile user has to satisfy its power constraint as

$$\sum_{i \in \mathcal{N}^{(t)}} P_{i,j}^{U,(t)} x_{i,j}^{(t)} \leq P_j^{max}, \ \forall j \in \mathcal{M}^{(t)}, \ \forall t \in \mathcal{T}, \tag{38}$$

where P_j^{max} is the maximal transmission power of mobile user j. Then, the upload link transmission time for task i from mobile user j can be calculated as

$$T_{i,j}^{U,(t)} = \frac{O_i^{(t)}}{R_{i,j}^{U,(t)}}. \tag{39}$$

Combine constraints (32) and (39), we have

$$e_{i,j}^{(t)} = P_{i,j}^{U,(t)} \times T_{i,j}^{U,(t)} + E_{i,j}^{exe,(t)}. \tag{40}$$

As for the download link transmission, the BS needs to transmit allocated tasks to selected mobile users. The transmission rate between the mobile user j and the BS for task i can be calculated as

$$R_{i,j}^{D,(t)} = w_{i,j}^{D,(t)} \log_2(1 + \gamma_{i,j}^{D,(t)}), \tag{41}$$

where $w_{i,j}^{D,(t)}$ is the allocated bandwidth for the download link channel, and $\gamma_{i,j}^{D,(t)} = \frac{P_{i,j}^{D,(t)} G_{i,j}^{D,(t)}}{\sigma^2}$ represents the SNR. $P_{i,j}^{D,(t)}$ and $G_{i,j}^{D,(t)}$ are the allocated transmission power and download link channel gain for task i to mobile user j, respectively. Similarly, the download link transmission time can be expressed as

$$T_{i,j}^{D,(t)} = \frac{S_i^{(t)}}{R_{i,j}^{D,(t)}}. \tag{42}$$

Since in wireless communication networks, the BS commonly has a transmission power limit

P_B^{max}, we have

$$\sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}} P_{i,j}^{D,(t)} x_{i,j}^{(t)} \le P_B^{max},\ \forall t \in \mathcal{T}. \tag{43}$$

With the consideration of the limited bandwidths for both the download and upload links, the following conditions should be imposed on the bandwidth allocation as

$$\sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}} w_{i,j}^{D,(t)} x_{i,j}^{(t)} \le B^D,\ \forall t \in \mathcal{T}, \tag{44}$$

$$\sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}} w_{i,j}^{U,(t)} x_{i,j}^{(t)} \le B^U,\ \forall t \in \mathcal{T}. \tag{45}$$

In order to meet the deadline requirement of each offloaded task, each computation task $J_i^{(t)}$ with input data size $S_i^{(t)}$ and output data size $O_i^{(t)}$ has to be transmitted and executed within $D_i^{(t)}$. This means we have the following constrain.

$$\sum_{j\in\mathcal{M}^{(t)}} (T_{i,j}^{D,(t)} + T_{i,j}^{exe,(t)} + T_{i,j}^{U,(t)}) x_{i,j}^{(t)} \le \min\{D_i^{(t)}, \Delta T\},\ \forall i \in \mathcal{N}^{(t)},\ \forall t \in \mathcal{T}. \tag{46}$$

Note that, as shown in (46), the total delay for each task is the summation of download transmission time, upload transmission time, and execution time. Moreover, since the download or upload transmission time is actually a ratio between the data size and the transmission rate, i.e., $R_{i,j}^{D,(t)}$ or $R_{i,j}^{U,(t)}$, the constraint (46) becomes nonlinear with respect to both transmission power and bandwidth. This requires us to consider a nonlinear online incentive mechanism design. However, designing such an online incentive mechanism is extremely challenging due to the nonlinearity and the multi-dimensional allocation outcome.

4.1.4 Utility of the BS

The utility of the BS consists of the benefit $r_{i,j}^{(t)}$ through offloading tasks to mobile users, the execution cost for the BS itself, the cost for the BS to transmit tasks to mobile users, and the rewards to mobile users, i.e., $\pi_{i,j}^{(t)}$. We mathematically formulate the utility of the BS as follows

$$U_B(\boldsymbol{x}^{(t)}) = \sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}} r_{i,j}^{(t)}x_{i,j}^{(t)} - \sum_{i\in\mathcal{N}^{(t)}}(1-\sum_{j\in\mathcal{M}^{(t)}} x_{i,j}^{(t)})\phi_i^{(t)}$$
$$-\sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}} P_{i,j}^{D,(t)}c_B^{(t)}x_{i,j}^{(t)} - \sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}}\pi_{i,j}^{(t)}, \tag{47}$$

where $\boldsymbol{x}^{(t)} = \{x_{1,1}^{(t)}, x_{1,2}^{(t)}, \cdots, x_{N^{(t)},M^{(t)}}^{(t)}\}$, $\phi_i^{(t)} = \theta_B Q_i^{(t)}$ is the energy consumption cost for the BS, and $c_B^{(t)}$ and θ_B are the transmission cost per unit power and energy cost per CPU cycle, respectively. $\pi_{i,j}^{(t)}$ denotes the reimbursement from the BS to mobile user j for task i. The BS can gain a benefit $r_{i,j}^{(t)}$ because the offloading process can, to large extent, preserve its own computation resource so that the BS will own enough resource for other intensive and complex applications.

4.1.5 Utility of Mobile User

The utility of any mobile user j at any time slot t consists of the payment from BS and the total cost related to executing task i. Thus, the utility of mobile user j can be formulated as

$$U_j(\boldsymbol{L}^{(t)}) = \sum_{i\in\mathcal{N}^{(t)}}(\pi_{i,j}^{(t)} - (P_{i,j}^{U,(t)}c_j^{(t)} + \theta_j Q_i^{(t)})x_{i,j}^{(t)}),\ \forall j\in\mathcal{M}^{(t)},\ \forall t\in\mathcal{T}, \tag{48}$$

where $\boldsymbol{L}^{(t)} = \{L_1^{(t)}, L_2^{(t)}, \cdots, L_{M^{(t)}}^{(t)}\}$ is a collection of all submitted bids at time slot t. Intuitively, since mobile users are intelligent and selfish, and their submitted information is private and unknown to the BS, they may submit their biding information strategically in order to earn more benefits in the competition. To this end, an incentive mechanism should be designed to force mobile users to bid their private information truthfully.

4.1.6 Problem Formulation

Before formulating our problem mathematically, we first introduce several properties that should be satisfied as follows.

- IC guarantees that no mobile users can gain more benefits by misreporting their private information. This property can be mathematically expressed as

$$U_j(L_j^{(t)}, \boldsymbol{L}_{-j}^{(t)}) \geq U_j(\acute{L}_j^{(t)}, \boldsymbol{L}_{-j}^{(t)}),\ \forall j \in \mathcal{M}^{(t)},\ \forall t \in \mathcal{T}, \tag{49}$$

where $\boldsymbol{L}_{-j}^{(t)}$ is the collection of bids excluding $L_j^{(t)}$, and $\acute{L}_j^{(t)}$ is a vector of potential false bids from mobile user j.

- IR ensures utilities of all mobile users are no less than zero, which can be formulated as

$$U_j(\boldsymbol{L}_j^{(t)}) \geq 0,\ \forall j \in \mathcal{M}^{(t)},\ \forall t \in \mathcal{T}. \tag{50}$$

- CR is defined as the ratio of the calculated online solutions over the optimal offline ones. Note that in this chapter, the upper bound of CR is one, and the larger CR is, the better performance becomes.

- CE is a metric that measures whether the designed online incentive mechanism can run in a polynomial time.

The BS has to jointly determine power allocations at both the BS and mobile users ($P_{i,j}^{D,(t)}, P_{i,j}^{U,(t)}$), upload and download link bandwidths ($w_{i,j}^{D,(t)}, w_{i,j}^{U,(t)}$), computation frequency ($f_{i,j}^{(t)}$), reimbursement to the mobile user ($\pi_{i,j}^{(t)}$), and task assignment ($x_{i,j}^{(t)}$) on-the-fly. The objective is to maximize the social welfare SW_{ob} of the system, which is defined as the summation of utilities from the BS and all mobile users, and can be calculated as

$$SW_{ob} = \sum_{t\in\mathcal{T}} U_B(\boldsymbol{x}^{(t)}) + \sum_{j\in\mathcal{M}^{(t)}} U_j(\boldsymbol{L}^{(t)}). \tag{51}$$

After some algebraic manipulations, the objective SW_{ob} can be equivalently rewritten as

$$\begin{aligned} SW_{ob} = &\sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}} a_{i,j}^{(t)} x_{i,j}^{(t)} - \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}} P_{i,j}^{D,(t)} c_B^{(t)} x_{i,j}^{(t)} \\ &- \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{N}^{(t)}}\sum_{j\in\mathcal{M}^{(t)}} P_{i,j}^{U,(t)} c_j^{(t)} x_{i,j}^{(t)} - \sum_{t\in\mathcal{T}}\sum_{i\in\mathcal{N}^{(t)}} \phi_i^{(t)}, \end{aligned} \tag{52}$$

where $a_{i,j}^{(t)} = r_{i,j}^{(t)} + \phi_i^{(t)} - \theta_j Q_i^{(t)}$. Note that since the last term in (52) is a constant, which cannot affect the optimality of the optimization problem, we will ignore this term in the following calculations and analysis.

Therefore, if all required information for all time slots can be obtained, the offline joint computation and communication resource allocation problem can be formulated as

$$
\begin{aligned}
&[\mathcal{P}1]: \\
&[\boldsymbol{X}, \boldsymbol{\Pi}] = \arg\max_{\boldsymbol{X}, \boldsymbol{\Pi}} SW_{ob} \\
&s.t. \quad (31), (34) - (36), (38), (43) - (46), (49) - (50), \\
&\qquad x_{i,j}^{(t)} \in \{0,1\}, w_{i,j}^{D,(t)} \geq 0, w_{i,j}^{U,(t)} \geq 0, P_{i,j}^{D,(t)} \geq 0, P_{i,j}^{U,(t)} \geq 0, \\
&\qquad f_{i,j}^{(t)} \geq 0, \pi_{i,j}^{(t)} \geq 0, \ \forall\, i \in \mathcal{N}^{(t)}, \forall\, j \in \mathcal{M}^{(t)}, \forall\, t \in \mathcal{T},
\end{aligned}
$$

where $\boldsymbol{X} = \{x_{i,j}^{(t)}, w_{i,j}^{D,(t)}, w_{i,j}^{U,(t)}, P_{i,j}^{D,(t)}, P_{i,j}^{U,(t)}, f_{i,j}^{(t)}\}$, and $\boldsymbol{\Pi} = \{\pi_{i,j}^{(t)}\}$, $\{i \in \mathcal{N}^{(t)}, j \in \mathcal{M}^{(t)}, t \in \mathcal{T}\}$ are decision variables. Obviously, it is very difficult, if not impossible, to solve $[\mathcal{P}1]$ in an online manner with a good CR because i) this problem is a typical mixed integer nonconvex optimization problem, which is well known to be NP-hard; ii) the constraint (36) in $[\mathcal{P}1]$ couples all time slots so that we cannot simply solve the problem for each time slot independently and unreasonable decisions in current time slot may affect future results; and iii) due to the consideration of transmission power for both download and upload links, constraints (36) and (46) become nonlinear and tightly coupled with each other, which further complicates the problem. Note that the above online mechanism design problem may be potentially solved by the MDP or the Lyapunov method. However, these two methods actually cannot be applied here because i) a *prior* distribution on biding information of mobile users, necessary for the MDP method, is not known in our case; ii) in the Lyapunov method, the objective should be an infinite time average form, while in this chapter, we consider any arbitrary time span. Moreover, only by carefully meeting certain requirements can the convergence of Lyapunov method be obtained. To this end, in the following sections, a novel online incentive mechanism with a sound CR is proposed.

4.2 Online Incentive Mechanism Design

In this section, we will present our online solution for the problem $[\mathcal{P}1]$, which consists of two steps. In the first step, by temporarily ignoring the energy constraint on each mobile user, an incentive mechanism, called One-shot Truthful Mechanism (OTM), will be designed for each single time slot. After that, we will rationally combine these independent single time slot solutions to design an online truthful mechanism.

4.2.1 One-shot Truthful Mechanism Design

In this subsection, we consider a single time slot and temporarily remove the energy constraint for each mobile user, i.e., the constraint (36). Hence, the original problem $[\mathcal{P}1]$ can be transformed into $[\mathcal{P}2]$ as follows. For the notional simplification, the superscript t in variables and sets are omitted in this subsection.

$$
\begin{aligned}
&[\mathcal{P}2]: \\
&\max_{\boldsymbol{X}} \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} a_{i,j}x_{i,j} - \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} P_{i,j}^{D} c_B x_{i,j} - \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} P_{i,j}^{U} c_j x_{i,j} \\
&s.t. \sum_{i\in\mathcal{N}} x_{i,j} \leq 1,\ \forall j \in \mathcal{M}, && (53) \\
&\quad \sum_{j\in\mathcal{M}} x_{i,j} \leq 1,\ \forall i \in \mathcal{N}, && (54) \\
&\quad \sum_{i\in\mathcal{N}} P_{i,j}^{U} x_{i,j} \leq P_j^{max},\ \forall j \in \mathcal{M}, && (55) \\
&\quad \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} P_{i,j}^{D} x_{i,j} \leq P_B^{max}, && (56) \\
&\quad \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} w_{i,j}^{D} x_{i,j} \leq B^{D}, && (57) \\
&\quad \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} w_{i,j}^{U} x_{i,j} \leq B^{U}, && (58) \\
&\quad \sum_{j\in\mathcal{M}} (T_{i,j}^{D} + T_{i,j}^{exe} + T_{i,j}^{U}) x_{i,j} \leq \min\{D_i, \Delta T\}, \forall i \in \mathcal{N} && (59) \\
&\quad T_{i,j}^{D} = \frac{S_i}{w_{i,j}^{D} x_{i,j} \log_2(1 + \frac{P_{i,j}^{D} x_{i,j} G_{i,j}^{D}}{\sigma^2})}, \\
&\quad T_{i,j}^{U} = \frac{O_i}{w_{i,j}^{U} x_{i,j} \log_2(1 + \frac{P_{i,j}^{U} x_{i,j} G_{i,j}^{U}}{\sigma^2})},\ T_{i,j}^{exe} = \frac{Q_i}{f_{i,j}}, \\
&\quad f_{i,j} \leq F_j,\ \forall j \in \mathcal{M}, && (60) \\
&\quad U_j(\boldsymbol{L}) \geq 0,\ \forall j \in \mathcal{M}, && (61) \\
&\quad U_j(L_j, \boldsymbol{L}_{-j}) \geq U_j(\acute{L}_j, \boldsymbol{L}_{-j}),\ \forall j \in \mathcal{M}. && (62)
\end{aligned}
$$

Moreover, since the reward or the payment, i.e., $\pi_{i,j}^{(t)}$, is not in the objective function of $[\mathcal{P}2]$, constraints (61) and (62), which respectively represent IR and IC, can be further removed. Note that this manipulation doesn't affect the optimality of $[\mathcal{P}2]$, and we will later reconsider constraints (61) and (62) by designing a payment rule. For clarity, we rewrite the newly formed optimization problem, i.e., $[\mathcal{RP}]$ as follows, which is termed as resource allocation problem in this chapter.

$$
\begin{aligned}
&[\mathcal{RP}]: \\
&\max_{\boldsymbol{X}} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}} a_{i,j} x_{i,j} - \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}} P^{D}_{i,j} c_B x_{i,j} - \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}} P^{U}_{i,j} c_j x_{i,j} \\
&s.t. \quad (53) - (60)
\end{aligned}
$$

Note that the problem $[\mathcal{RP}]$ is still NP-hard due to the non-convexity and mixed integer optimization.

Generally, randomized rounding algorithms [109] are a potential solutions for problem $[\mathcal{RP}]$, which commonly consists of a relaxation algorithm $\mathcal{A}$ and a rounding algorithm $\mathcal{D}(\dot{\boldsymbol{X}})$. $\mathcal{A}$ is used to obtain the fractional solution of $[\mathcal{RP}]$, i.e., $\dot{\boldsymbol{X}}$, while $\mathcal{D}(\dot{\boldsymbol{X}})$ maps $\dot{\boldsymbol{X}}$ to integer solutions based on a predefined rounding scheme, such as poisson rounding scheme [110]. Unfortunately, as shown in [111], the outcomes from randomized rounding algorithms could hardly meet constraints of IR and IC. Instead, in this chapter, by considering the extreme complexity of $[\mathcal{RP}]$ and motivated by the Maximal-in-Distributional Range (MIDR) rule [112], we propose a new method, called Integrate Rounding Scheme based MIDR (IRSM), for the one-shot truthful mechanism.

The basic idea of our proposed IRSM can be explained as follows. In IRSM, we integrate the integer solution $\boldsymbol{X}$ and the rounding scheme $\mathcal{D}(\dot{\boldsymbol{X}})$ into the objective function $f(\boldsymbol{X}, \boldsymbol{L})$, and then find a fractional solution $\dot{\boldsymbol{X}}$ that optimizes the expected objective function $\mathbb{E}_{\boldsymbol{X} \sim \mathcal{D}(\dot{\boldsymbol{X}})}\{f(\boldsymbol{X}, \boldsymbol{L})\}$ among all feasible fractional solutions. After that, the rounding scheme, i.e., $\mathcal{D}(\dot{\boldsymbol{X}})$, is used to obtain the integer solutions $\boldsymbol{X}$ from fractional solutions $\dot{\boldsymbol{X}}$. Although this optimization problem is usually intractable due to the embedding of the rounding function into the objective function, we can still manage it in this chapter. Note that there exists a natural distinction between our proposed IRSM and the traditional randomized rounding algorithm. In fact, our IRSM is to integrate the rounding scheme in the objective and directly optimize the integer variable $\boldsymbol{X}$, rather than the relaxed original problem, which is optimized in the traditional randomized rounding algorithm. **Algorithm** 7 presents the steps of our proposed IRSM algorithm.

Note that for most rounding schemes in the literature, the expected maximization problem in the

Algorithm 7: The framework of our proposed IRSM algorithm.

Input: Submitted biddings $\boldsymbol{L}$.

Output: Feasible solution $\boldsymbol{X}$.

1 Obtain the optimal fractional solution $\dot{\boldsymbol{X}}^*$ by maximizing $\mathbb{E}_{\boldsymbol{X}\sim\mathcal{D}(\dot{\boldsymbol{X}})}\{f(\boldsymbol{X},\boldsymbol{L})\}$;

2 Rounding the final solution $\boldsymbol{X}$ based on the rounding scheme $\mathcal{D}(\dot{\boldsymbol{X}}^*)$.

Algorithm 7 cannot be solved in polynomial time [110]. Besides, in order to apply **Algorithm** 7 to design OTM, this expected maximization problem should be solved optimally. To this end, in this chapter, the widely used rounding scheme, i.e., poisson rounding scheme, is applied in our analyses, which can well balance the computational complexity and the achievable performance. We will prove later that by applying this rounding scheme, the designed OTM can achieve an approximation ratio of $1-\frac{1}{e}$ with polynomial running time, where e, known as the Euler's number, is a mathematical constant. Note that the proposed framework, i.e., **Algorithm** 7, can be applied to other rounding schemes.

Therefore, we reformulate $[\mathcal{RP}]$ by taking the following two operations:

- Relax the integer variable, i.e., $x_{i,j}$, to be a real variable, i.e., $\dot{x}_{i,j}$, between 0 and 1;
- Take the expectation on the objective function in the $[\mathcal{RP}]$ based on the poisson rounding scheme $\mathcal{D}(\dot{\boldsymbol{X}})$.

Then, the problem $[\mathcal{RP}]$ can be reformulated

$$\begin{aligned}
&[\mathcal{RPO}]:\\
&\max_{\dot{\boldsymbol{X}}}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\mathbb{E}_{\boldsymbol{X}\sim\mathcal{D}(\dot{\boldsymbol{X}})}\{(a_{i,j}-P^D_{i,j}\varpi_{B,j}-P^U_{i,j}\varpi_j)x_{i,j}\}\\
&s.t.\quad (53)-(60).
\end{aligned}$$

where $\dot{\boldsymbol{X}}$ is the set of fractional variables of $[\mathcal{RPO}]$, and $\varpi_{B,j}$ and ϖ_j are virtual costs related to c_B and $c_{i,j}$, respectively, which will be further explained in subsection 4.2.2. Define $\boldsymbol{P}^D$ and $\boldsymbol{P}^U$ as the sets of $P^D_{i,j}$ and $P^U_{i,j}$, respectively, and let $\tilde{P}^U_{i,j}=P^U_{i,j}x_{i,j}$, $\tilde{P}^D_{i,j}=P^D_{i,j}x_{i,j}$, $\tilde{w}^U_{i,j}=w^U_{i,j}x_{i,j}$, and

$\tilde{w}_{i,j}^{D} = w_{i,j}^{D} x_{i,j}$. By applying the poisson rounding scheme in $[\mathcal{RPO}]$, we have

$$
\begin{aligned}
&[\mathcal{ERPO}]: \\
&\max_{\dot{\boldsymbol{X}}} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} (a_{i,j} - \tilde{P}_{i,j}^{D} \varpi_{B,j} - \tilde{P}_{i,j}^{U} \varpi_j)(1 - e^{-x_{i,j}}) \\
&s.t. \quad (53), \quad (54), && (63)\\
&\quad \sum_{i\in\mathcal{N}} \tilde{P}_{i,j}^{U} \leq P_j^{max},\ \forall j \in \mathcal{M},\ \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} \tilde{P}_{i,j}^{D} \leq P_B^{max}, && (64)\\
&\quad \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} \tilde{w}_{i,j}^{D} \leq B^{D}, \quad \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} \tilde{w}_{i,j}^{U} \leq B^{U}, && (65)\\
&\quad \sum_{j\in\mathcal{M}} (\tilde{T}_{i,j}^{D} + T_{i,j}^{exe} + \tilde{T}_{i,j}^{U}) x_{i,j} \leq \min\{D_i, \Delta T\}, && (66)\\
&\quad \tilde{T}_{i,j}^{D} = \frac{S_i}{\tilde{w}_{i,j}^{D} \log_2(1 + \frac{\tilde{P}_{i,j}^{D,t} G_{i,j}^{D}}{\sigma^2})},\ T_{i,j}^{exe} = \frac{Q_i}{f_{i,j}},\ f_{i,j} \leq F_j, && (67)\\
&\quad \tilde{T}_{i,j}^{U} = \frac{O_i}{\tilde{w}_{i,j}^{U} \log_2(1 + \frac{\tilde{P}_{i,j}^{U} G_{i,j}^{U}}{\sigma^2})}. && (68)
\end{aligned}
$$

Obviously, even though $x_{i,j}$ is fractional, it is still challenging to solve $[\mathcal{ERPO}]$ because both the objective and the constraint (66) are non-convex. To address this issue, we introduce the following substitutes:

$$
\begin{aligned}
&x_{i,j} = -\chi_{i,j}^{3}, && (69)\\
&\ln(1 + \frac{\tilde{P}_{i,j}^{D,t} G_{i,j}^{D}}{\sigma^2}) = \alpha_{i,j}^{D}, && (70)\\
&\ln(1 + \frac{\tilde{P}_{i,j}^{U,t} G_{i,j}^{U}}{\sigma^2}) = \alpha_{i,j}^{U}, && (71)\\
&e^{\alpha_{i,j}^{D} + \chi_{i,j}^{3}} = z_{i,j}^{D}, && (72)\\
&e^{\alpha_{i,j}^{U} + \chi_{i,j}^{3}} = z_{i,j}^{U}. && (73)
\end{aligned}
$$

Then, the optimization problem $[\mathcal{ERPO}]$ can be reformulated as

$$
\begin{aligned}
&[\mathcal{ERPO}1]:\\
&\max \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} a_{i,j}(1-e^{\chi^3_{i,j}})-\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} b_{i,j}(e^{\alpha^D_{i,j}}+e^{\chi^3_{i,j}}-z^D_{i,j}-1)\\
&\qquad\qquad -\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}} d_{i,j}(e^{\alpha^U_{i,j}}+e^{\chi^3_{i,j}}-z^U_{i,j}-1)\\
&s.t. -\sum_{i\in\mathcal{N}}\chi^3_{i,j}\le 1,\ \forall j\in\mathcal{M},\quad -\sum_{j\in\mathcal{M}}\chi^3_{i,j}\le 1,\ \forall i\in\mathcal{N}, &(74)\\
&\quad \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\tilde{w}^D_{i,j}\le B^D,\quad \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\tilde{w}^U_{i,j}\le B^U, &(75)\\
&\quad \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}\frac{\sigma^2}{G^D_{i,j}}(e^{\alpha^D_{i,j}}-1)\le P^{max}_B, &(76)\\
&\quad \sum_{i\in\mathcal{N}}\frac{\sigma^2}{G^U_{i,j}}(e^{\alpha^U_{i,j}}-1)\le P^{max}_j,\ \forall j\in\mathcal{M}, &(77)\\
&\quad \sum_{j\in\mathcal{M}}\Big(\frac{\chi^3_{i,j}S_i}{\tilde{w}^D_{i,j}\alpha^D_{i,j}\log_2 e}+\frac{\chi^3_{i,j}O_i}{\tilde{w}^U_{i,j}\alpha^U_{i,j}\log_2 e}+\frac{\chi^3_{i,j}Q_i}{f_{i,j}}\Big)\ge -\min\{D_i,\Delta T\}, &(78)\\
&\quad f_{i,j}\le F_j,\ \forall j\in\mathcal{M}, &(79)\\
&\quad (72)-(73),\\
&\quad \chi_{i,j}\in[-1,0],\alpha^D_{i,j}\ge 0,\alpha^U_{i,j}\ge 0,\tilde{w}^D_{i,j}\ge 0,\tilde{w}^U_{i,j}\ge 0,\\
&\qquad z^D_{i,j}\ge 0, z^U_{i,j}\ge 0, f_{i,j}\ge 0,
\end{aligned}
$$

where $b_{i,j}=\frac{\sigma^2\varpi_{B,j}}{G^D_{i,j}}$ and $d_{i,j}=\frac{\sigma^2\varpi_j}{G^U_{i,j}}$.

Lemma 12. *The transformed optimization problem $[\mathcal{ERPO}1]$ is a convex optimization problem.*

Proof. Please refer to Appendix C.1. ■

Since the optimization problem $[\mathcal{ERPO}1]$ is convex, we can solve it optimally by the Lagrange dual technique [86]. Denote the outcomes of problem $[\mathcal{EPRO}]$ as $\dot{\boldsymbol{X}}^*=\{x^*_{i,j},w^{D*}_{i,j},w^{U*}_{i,j},P^{D*}_{i,j},P^{U*}_{i,j},f^*_{i,j}\}$, and the final result is determined as $\boldsymbol{X}=\{x_{i,j},w^D_{i,j},w^U_{i,j},P^D_{i,j},P^U_{i,j},f_{i,j}\}$ by adopting poisson rounding scheme.

After solving the resource allocation problem, we move to design a payment scheme $\pi_{i,j}$ to satisfy constraints (61) and (62). Note that unlike existing work [33, 113] where allocation result was single-dimensional so that the Myerson lemma can be directly applied. In this chapter, the allocation outcome is six-dimensional, including upload and download link power controls, upload and download link bandwidth allocations, task assignment, and computation resource allocation. Thus, we design a new payment scheme based on the framework of VCG mechanism. The designed reward or the payment scheme for mobile user j for the execution of task i is

$$\pi_{i,j} = \frac{\sum_{i \in \mathcal{N}} (P^{U}_{i,j} \varpi_j + \theta_j Q_i) x_{i,j}}{\sum_{i \in \mathcal{N}} (P^{U*}_{i,j} \varpi_j + \theta_j Q_i)(1 - e^{-x^*_{i,j}})} \pi^f_{i,j}, \ \forall j \in \mathcal{U}^{sel},$$

where $\mathcal{U}^{sel}$ is the set of selected mobile users, and $\pi^f_{i,j}$ is defined as the fractional payment, which can be calculated as

$$\begin{aligned} \pi^f_{i,j} = & \sum_{i \in \mathcal{N}} \sum_{j' \in \mathcal{M}/\{j\}} (a_{i,j'} - P^{D*}_{i,j'} \varpi_{B,j'} - P^{U*}_{i,j'} \varpi_{j'})(1 - e^{-x^*_{i,j'}}) \\ & - \max_{\boldsymbol{X}} \sum_{i \in \mathcal{N}} \sum_{j' \in \mathcal{M}/\{j\}} (a_{i,j'} - P^{D}_{i,j'} \varpi_{B,j'} - P^{U}_{i,j'} \varpi_{j'})(1 - e^{-x_{i,j'}}). \end{aligned}$$

Till now, the whole design process of OTM for solving $[\mathcal{P}2]$ has been completed. For better understanding, we summarize the whole procedures of OTM in **Algorithm** 8.

Lemma 13. *The approximation ratio of the proposed* ***Algorithm*** *8 is* $1 - \frac{1}{e}$ *in expectation.*

Proof. We define SW_{ob} as the objective value by the **Algorithm** 8, and $x_{i,j}$, $P^{D}_{i,j}$, and $P^{U}_{i,j}$ are corresponding solutions. Moreover, let SW^*_{ob} be the optimal objective value of the problem $[\mathcal{RP}]$.

Algorithm 8: One-shot truthful mechanism by proposed IRSM.

```
Initialization;
U^sel = ∅;
x_{i,j} = 0, ∀ i ∈ N, ∀ j ∈ M;
Solve the convex problem [ERPO1] to obtain fraction solutions
    x*_{i,j}, w^{D*}_{i,j}, w^{U*}_{i,j}, P^{D*}_{i,j}, P^{U*}_{i,j}, f*_{i,j};
while i ∈ N do
    Draw d_i uniformly at random from [0, 1];
    if Σ_{j∈(M/U)} (1 − e^{−x*_{i,j}}) ≥ d_i then
        Let j* be the minimum index to satisfy that Σ_{j≤j*} (1 − e^{−x*_{i,j}}) ≥ d_i;
        U^sel = U^sel ∪ j*;
        x_{i,j*} = 1 ;
Make the payment to selected mobile user j as π_{i,j} = [Σ_{i∈N} (P^U_{i,j} ϖ_j + θ_j Q_i) x_{i,j}] / [Σ_{i∈N} (P^{U*}_{i,j} ϖ_j + θ_j Q_i)(1 − e^{−x*_{i,j}})] π^f_{i,j};
Output;
x_{i,j}, w^D_{i,j}, w^U_{i,j}, P^D_{i,j}, P^U_{i,j}, f_{i,j}, π_{i,j} ;
```

Then, we have

$$
\begin{aligned}
&\mathbb{E}\{SW_{ob}\} = \mathbb{E}\{\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}(a_{i,j} - P^D_{i,j}\varpi_{B,j} - P^U_{i,j}\varpi_j)x_{i,j}\}\\
&= \sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}(a_{i,j} - P^D_{i,j}\varpi_{B,j} - P^U_{i,j}\varpi_j)(1 - e^{-x_{i,j}})\\
&\geq (\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}(a_{i,j} - P^{D*}_{i,j}\varpi_{B,j} - P^{U*}_{i,j}\varpi_j)(1 - e^{-x^*_{i,j}}))\\
&\geq (1-\frac{1}{e})\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}(a_{i,j} - P^{D*}_{i,j}\varpi_{B,j} - P^{U*}_{i,j}\varpi_j)x^*_{i,j}\\
&\geq (1-\frac{1}{e})SW^*_{ob}.
\end{aligned}
$$

This completes the proof. ■

Lemma 14. *The proposed mechanism (OTM) is individually rationale in expectation.*

Proof. We have

$$\begin{aligned}
\mathbb{E}\{U_j(\boldsymbol{L})\} &= (1 - e^{-x^*_{i,j}})(\pi_{i,j} - \sum_{i\in\mathcal{N}}(P^{U*}_{i,j}\varpi_j + \theta_j Q_i)x_{i,j}) \\
&= \frac{(1-e^{-x^*_{i,j}})\sum_{i\in\mathcal{N}}(P^{U}_{i,j}\varpi_j+\theta_j Q_i)}{\sum_{i\in\mathcal{N}}(P^{U*}_{i,j}\varpi_j+\theta_j Q_i)(1-e^{-x^*_{i,j}})} \times (\pi^f_{i,j} - \sum_{i\in\mathcal{N}}(P^{U*}_{i,j}\varpi_j+\theta_j Q_i)(1-e^{-x^*_{i,j}})) \\
&= \pi^f_{i,j} - \sum_{i\in\mathcal{N}}(P^{U*}_{i,j}\varpi_j + \theta_j Q_i)(1 - e^{-x^*_{i,j}}) \\
&= U^f_j,
\end{aligned} \tag{80}$$

where U^f_j is defined as the fractional utility of mobile user j. In the following, we only need to prove that the fractional utility, i.e., U^f_j, is no less than zero. Based on our proposed payment design, U^f_j can be further calculated as

$$\begin{aligned}
U^f_j &= \pi^f_{i,j} - (P^{U*}_{i,j}\varpi_j + \theta_j Q_i)(1 - e^{-x^*_{i,j}}) \\
&= \sum_{i\in\mathcal{N}}\sum_{j'\in\mathcal{M}/\{j\}}(a_{i,j'} - P^{D*}_{i,j'}\varpi_{B,j'} - P^{U*}_{i,j'}\varpi_{j'})(1 - e^{-x^*_{i,j'}}) \\
&-\max_{\boldsymbol{X}}\sum_{i\in\mathcal{N}}\sum_{j'\in\mathcal{M}/\{j\}}(a_{i,j'} - P^{D}_{i,j'}\varpi_{B,j'} - P^{U}_{i,j'}\varpi_{j'})(1-e^{-x_{i,j'}}) - (P^{U*}_{i,j}\varpi_j + \theta_j Q_i)(1 - e^{-x^*_{i,j}}) \\
&= \max_{\boldsymbol{X}}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}}(a_{i,j} - P^{D*}_{i,j}\varpi_{B,j} - P^{U*}_{i,j}\varpi_j)(1 - e^{-x^*_{i,j}}) \\
&-\max_{\boldsymbol{X}}\sum_{i\in\mathcal{N}}\sum_{j'\in\mathcal{M}/\{j\}}(a_{i,j'} - P^{D}_{i,j'}\varpi_{B,j'} - P^{U}_{i,j'}\varpi_{j'})(1-e^{-x_{i,j'}}) \\
&\geq 0,
\end{aligned} \tag{81}$$

where the last inequality holds because the first term in (81) is the maximal social welfare with total N mobile users, while the second term in (81) is the maximal social welfare excluding mobile user j. Thus, the fractional payment is individually rational, so is $\mathbb{E}\{U_j(\boldsymbol{L})\}$. This completes the proof. ■

Lemma 15. *The proposed mechanism (OTM) is incentively compatible (truthful) in expectation.*

Proof. We assume that mobile user j misreports its actual private information ϖ_j as $\acute{\varpi}_j$, and the fractional solutions based on this false value are $\acute{x}_{i,j}$, $\acute{P}^D_{i,j}$, and $\acute{P}^U_{i,j}$. Then, the fractional utility due to the falsely reported information can be calculated as

$$\acute{U}^f_j = \max_{\boldsymbol{X}} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} (a_{i,j} - \acute{P}^D_{i,j}\varpi_{B,j} - \acute{P}^U_{i,j}\varpi_j)(1 - e^{-\acute{x}_{i,j}})$$
$$-\max_{\boldsymbol{X}} \sum_{i\in\mathcal{N}} \sum_{j'\in\mathcal{M}/\{j\}} (a_{i,j'} - P^D_{i,j'}\varpi_{B,j'} - P^U_{i,j'}\varpi_{j'})(1 - e^{-x_{i,j'}})$$

The difference between U^f_j and $\acute{U}^f_j$ can be calculated as

$$\begin{aligned} U^f_j - \acute{U}^f_j = &\max_{\boldsymbol{X}} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} (a_{i,j} - P^{D*}_{i,j}\varpi_{B,j} - P^{U*}_{i,j}\varpi_j)(1 - e^{-x^*_{i,j}}) \\ &- \max_{\boldsymbol{X}} \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{M}} (a_{i,j} - \acute{P}^D_{i,j}\varpi_{B,j} - \acute{P}^U_{i,j}\varpi_j)(1 - e^{-\acute{x}_{i,j}}) \\ &\geq 0. \end{aligned} \tag{82}$$

Since the first term in inequality (82) is the maximal objective under the true values of ϖ_j and $\varpi_{B,j}$, $U^f_j - \acute{U}^f_j \geq 0$ means the fractional payment is truthful. As a result, by combing (80), our proposed mechanism is also truthful in expectation. This completes the proof. ■

Theorem 6. *The proposed mechanism (OTM) is $1 - \frac{1}{e}$-approximation, truthful and individual rationality in expectation.*

Proof. The conclusion can be proved by combining the above **Lemmas**. ■

In the next subsection, we will reconsider the constraint (36), which is the energy budget constraint for each mobile user, and adopt our designed OTM to devise an online truthful mechanism.

4.2.2 Online Truthful Mechanism Design

Since the energy constraint for each mobile user is applied to all time slots, some mobile users may use up their energies before T time slots, so that they cannot participate the competition campaign in the rest time slots. This may result in a lower objective value because it narrows down the possible choosing space of mobile users, and the remaining mobile users may submit valuation, i.e., $c^{(t)}_j$ with high values in the future time slots. To address this issue, we intentionally extend lifetime of mobile users to allow as many mobile users as possible be survived in any time slot, so as to allow the BS to explore more cost-efficient mobile users for increasing the objective value. Following this intuition, we establish a relationship between the remained energy and the total energy, and intentionally decrease the mobile users' winning probabilities in the upcoming time slots if they have been selected before.

Specifically, we introduce two auxiliary variables, i.e., $\nu_j^{(t)}$ and $\nu_{B,j}^{(t)}$ for each mobile user j, and increase their values from an initial value $\frac{1}{\varphi}$, where $\varphi = \max_{t \in \mathcal{T}, j \in \mathcal{U}^{sel,(t)}} \frac{e_{i_j,j}^{(t)}}{E_j}$, denotes the maximum ratio between the energy consumption and the total energy. Obviously, φ should be much less than 1 as no mobile user is willing to use too much energy within a single time slot. Moreover, instead of applying actual unit power cost $c_j^{(t)}$ and $c_B^{(t)}$ in the **Algorithm** 8, we replace them by $\varpi_j^{(t)} = \beta_j c_j \varphi \nu_j^{(t-1)}$ and $\varpi_{B,j}^{(t)} = \beta_j c_B \varphi \nu_{B,j}^{(t-1)}$, where β_j ($\beta_j > 1$), called the inflating factor, is used to improve the performance of our online truthful mechanism, and $\beta = \max_{j \in \mathcal{M}} \beta_j$. Note that $\nu_j^{(t-1)}$ and $\nu_{B,j}^{(t-1)}$ are both updated carefully to ensure that the mobile user who has lower remaining energy will have a less chance to be selected in future time slots. For convenience, we summarize the online truthful mechanism as in the **Algorithm** 9. It is worth noting that by using the updating manner on $\nu_j^{(t-1)}$ and $\nu_{B,j}^{(t-1)}$ as in **Algorithm** 9, theoretical CR exists. Moreover, unlike [33, 113], where the formulated problems were linear, in this chapter, we consider a nonlinear scenario, but still having a CR guarantee.

Algorithm 9: Online truthful mechanism.

Initialization;

$\nu_j^{(0)} = \nu_{B,j}^{(0)} = \frac{1}{\varphi}, \ \forall j \in \mathcal{M}$;

for *each time slot* $t \in \mathcal{T}$ **do**

- $\varpi_j^{(t)} = \beta c_j^{(t)} \varphi \nu_j^{(t-1)}$;
- $\varpi_{B,j}^{(t)} = \beta c_B^{(t)} \varphi \nu_{B,j}^{(t-1)}$;
- Run **Algorithm** 8 to get the set of selected mobile users and their indexes of execution tasks, which are denoted as $\mathcal{U}^{sel,(t)}$ and i_j, respectively;
- Based on (40), calculate energy consumptions $e_{i_j,j}^{(t)}$ at mobile user $j \in \mathcal{U}^{sel,(t)}$;
- $\nu_j^{(t)} = \nu_j^{(t-1)}(1 + \frac{e_{i_j,j}^{(t)}}{E_j}) + \frac{e_{i_j,j}^{(t)}}{\varphi E_j}, \ j \in \mathcal{U}^{sel,(t)}$;
- $\nu_{B,j}^{(t)} = \nu_{B,j}^{(t-1)}(1 + \frac{e_{i_j,j}^{(t)}}{E_j}) + \frac{e_{i_j,j}^{(t)}}{\varphi E_j}, \ j \in \mathcal{U}^{sel,(t)}$;
- $\nu_j^{(t)} = \nu_j^{(t-1)}, \nu_{B,j}^{(t)} = \nu_{B,j}^{(t-1)}, \ j \in \mathcal{M}/\mathcal{U}^{sel,(t)}$;

Lemma 16. *The CR of the proposed online truthful mechanism is* $\beta(1-\frac{1}{e})(2^{\varphi}-1)$ *in expectation.*

Proof. Let's begin with the proof of the following inequalities

$$\nu_j^{(t)} \approx \frac{1}{\varphi}(2^{\frac{\sum_{t\in\mathcal{T}} e_{i_j,j}^{(t)}}{E_j}} - 1) \le \frac{1}{\varphi}(2^{\varphi}-1), \tag{83}$$

$$\nu_{B,j}^{(t)} \approx \frac{1}{\varphi}(2^{\frac{\sum_{t\in\mathcal{T}} e_{i_j,j}^{(t)}}{E_j}} - 1) \le \frac{1}{\varphi}(2^{\varphi}-1). \tag{84}$$

We prove inequalities (83) and (84) by induction on time index, t. Initially, we assume those inequalities are all held at time slot $t-1$, and the task i will be assigned to the mobile user j at time slot t. By taking inequality (83) as an example, we have

$$\begin{aligned}
\nu_j^{(t)} &= \nu_j^{(t-1)}(1+\frac{e_{i_j,j}^{(t)}}{E_j}) + \frac{e_{i_j,j}^{(t)}}{\varphi E_j}, \\
&\approx \frac{1}{\varphi}(2^{\frac{\sum_{t\in\mathcal{T}/t} e_{i,j}^{(t)}}{E_j}} - 1)(1+\frac{e_{i_j,j}^{(t)}}{E_j}) + \frac{e_{i_j,j}^{(t)}}{\varphi E_j}, \\
&= \frac{1}{\varphi}(2^{\frac{\sum_{t\in\mathcal{T}/t} e_{i,j}^{(t)}}{E_j}}(1+\frac{e_{i_j,j}^{(t)}}{E_j}) - 1), \\
&\approx \frac{1}{\varphi}(2^{\frac{\sum_{t\in\mathcal{T}} e_{i_j,j}^{(t)}}{E_j}} - 1) \le \frac{1}{\varphi}(2^{\varphi}-1),
\end{aligned} \tag{85}$$

where the approximation in (85) holds because $2^x \approx 1+x$ when x is small. Let $\overline{P}^{(t)}$ be the optimal objective value of the problem $[\mathcal{RP}]$ at time slot t without the consideration of the constraint (36), and $P^{(t)*}$ is the optimal objective value at time slot t calculated by the **Algorithm** 8. We have

$$\begin{aligned}
P^{(t)*} &= \sum_{j\in\mathcal{U}^*}(a_{i_j,j} - P_{i_j,j}^{D*}\varpi_{B,j} - P_{i_j,j}^{U*}\varpi_j) \\
&\ge \sum_{j\in\overline{\mathcal{U}}}(a_{i_j,j} - \overline{P}_{i_j,j}^{D}\varpi_{B,j} - \overline{P}_{i_j,j}^{U}\varpi_j) \\
&\ge \beta(2^{\varphi}-1)\sum_{j\in\overline{\mathcal{U}}}(a_{i_j,j} - \overline{P}_{i_j,j}^{D}c_B - \overline{P}_{i_j,j}^{U}c_j) \\
&= \beta(2^{\varphi}-1)\overline{P}^{(t)}.
\end{aligned} \tag{86}$$

Therefore, the objective value $\hat{P}^{(t)}$ from the **Algorithm** 9 at time slot t can be calculated as

$$\begin{aligned}\hat{P}^{(t)} &= \sum_{j\in\mathcal{U}^{sel,(t)}} (a_{i_j,j} - P^D_{i_j,j}c_B - P^U_{i_j,j}c_j) \\ &\geq \sum_{j\in\mathcal{U}^{sel,(t)}} (a_{i_j,j} - P^D_{i_j,j}\varpi_{B,j} - P^U_{i_j,j}\varpi_j) \qquad (87) \\ &\geq (1-\frac{1}{e})P^{(t)*} \\ &\geq \beta(1-\frac{1}{e})(2^{\varphi}-1)\overline{P}^{(t)},\end{aligned}$$

where inequality (87) holds since $c_j \leq \varpi_j$ and $c_B \leq \varpi_{B,j}$. Summing over total time slots, we have

$$\begin{aligned}\hat{P} &= \sum_{t\in\mathcal{T}} \hat{P}^{(t)} \geq \beta(1-\frac{1}{e})(2^{\varphi}-1)\sum_{t\in\mathcal{T}}\overline{P}^{(t)} \\ &\geq \beta(1-\frac{1}{e})(2^{\varphi}-1)P^{opt}, \qquad (88)\end{aligned}$$

where P^{opt} is the optimal objective value of the problem $[\mathcal{RP}]$, and the last inequality holds because $\sum_{t\in\mathcal{T}}\overline{P}^{(t)}$ is the summation of all independent time slots without the consideration of the constraint (36), which is no less than that in our proposed scenario. This completes the proof. ■

Lemma 17. *Our proposed online truthful mechanism can obtain the result in polynomial time.*

Proof. In this chapter, the computational complexity of the proposed online mechanism is evaluated in terms of computation times with respect to the number of offloading tasks, mobile users, and total time slots. Remind that our proposed online truthful mechanism consists of two parts, **Algorithm** 8 and **Algorithm** 9. As for **Algorithm** 8, we solve the fractional convex problem $[\mathcal{EPRO}1]$ by gradient descent optimizing solver. Suppose that L_{max} and I_{max} are the maximal iteration numbers to solve the Lagrangian primal problem and its primal problem, respectively. Then, the computational complexity of **Algorithm** 8 can be calculated as $O(NM + L_{max}I_{max})$. For the **Algorithm** 9, the total computational complexity is $O(|\mathcal{T}| \times (NM + L_{max}I_{max}))$. In summary, the proposed online truthful mechanism runs in polynomial time, which completes the proof. ■

Theorem 7. *The proposed online truthful mechanism is truthful and individually rational, and the*

CR is $\beta(1 - \frac{1}{e})(2^{\varphi} - 1)$ *in expectation.*

Proof. The conclusion can be proved by combining the **Lemma** 16 and the **Theorem** 6. ■

4.3 Numerical Results

In this section, numerical simulations are conducted to verify the effectiveness of our proposed online truthful mechanism. Since the total social welfare, and the utility of mobile users are the most important economical metrics and the competitive ratio is vital to measure an online truthful mechanism, we will focus on evaluating these three performance metrics with respect to different numbers of offloading tasks and mobile users. In the simulations, the wireless channels between the BS and mobile users experience Rayleigh fading and all channel coefficients are zero-mean, CSCG random variables with variances $d^{-\frac{v}{2}}$, where d is the distance between the transmitter and the receiver and $v = 4$. Table 4.2 lists the main simulation parameter values, most of which have been employed in [55, 57, 71, 108, 114]. In the following figures, each performance point is derived by averaging 300 independent runs. For comparison purpose, the following three benchmarks online strategies are simulated as well.

- Lyapunov based online mechanism (LOM): The Lyapunov optimization was used in [57] for designing online mechanism, but working at a fixed maximal upload transmission power. Note that since the objective by using the Lyapunov method should be a long-term one, we only calculate the first $|\mathcal{T}|$ time slots in the simulation.

- Fixed power online mechanism (FPOM): The transmission power of both upload and download links are fixed in advance, and the online mechanism problem is solved by our proposed method.

- Random online mechanism (ROM): In each time slot, the optimization variables, i.e., $\boldsymbol{X}$, are determined randomly, and mobile users will not be excluded from future participation until their energy budgets deplete.

Fig. 4.2 illustrates the total social welfare obtained by different online mechanisms with respect to different number of mobile users when the tasks offloaded by the BS are randomly chosen from

Table 4.2: Main Simulation Parameters In Chapter 4

Parameter	Value
Cell radius	500 m
Download link bandwidth	40 MHz
Upload link bandwidth	10 MHz
P_B^{max} at the BS	46 dBm
P_j^{max} at mobile users	23 dBm
Background noise average power	-60 dBm
Total running time	20 minutes
Time slot length	60 seconds
Input task size	Randomly from 10 to 30 MB
Output task size	20% of the input data
CPU cycles coefficient	330 cycles/Byte
β_j at mobile users	10^{-26}
θ_B at the BS	$\$10^{-10}$
$c_B^{(t)}$ at the BS	$0.1
Benefits, i.e., $r_{i,j}^{(t)}$, at the BS	Randomly over (1, 2]
θ_j at mobile users	$\$0.5 \times 10^{-10}$
$c_j^{(t)}$ at mobile users	Randomly from $[\$0.5, \$1]$
$D_i^{(t)}$ delay demand	Randomly from [1, 15] seconds
F_j at mobile users	2 GHz

[10, 20] for each time slot, and the energy budget is set to 8×10^3 (Joule). It can be seen from this figure that the total social welfare by all online mechanisms increases with the number of mobile users till reaching saturation when the number of mobile users is large enough. This can be explained as follows. With the number of mobile users increasing, more and more tasks can be successfully offloaded to mobile users so that the total social welfare increases. However, with the excessive amount of mobile users, e.g., 30, all tasks offloaded by the BS are accepted and executed by winning mobile users, and no extra mobile users can contribute to the total social welfare. Moreover, our proposed online mechanism is superior to all other three online mechanisms. This is because our proposed method jointly optimizes upload and download links' transmission powers, while the LOM ignores the optimization of upload link transmission power, and the FPOM doesn't consider the power control. Furthermore, Fig. 4.3 reevaluates the relationship between the total social welfare and a various number of mobile users when there are 18 tasks needing to be offloaded in each time slot. From this figure, we can almost draw the same conclusions as those

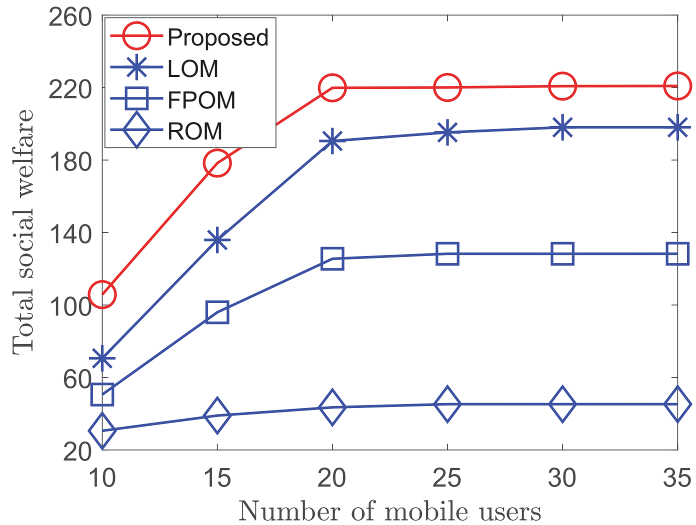

Figure 4.2: Total social welfare versus number of mobile users when offloaded tasks are with different numbers in each time slot.

in Fig. 4.3. In addition, it is worth noting that when the number of mobile users is large enough, the performance by the LOM is close to our proposed one. The reason behind could be that more mobile users mean that more potential suitable mobile users (e.g., larger $r_{i,j}$ and lower bidding cost $c_{i,j}^{(t)}$) can be selected by the BS, which, to some extent, can reduce the effect of upload link power control.

Fig. 4.4 reveals the total social welfare under different numbers of arriving tasks in each time slot. In the simulation, the total number of mobile users is 18, i.e., $|\mathcal{M}| = 18$, and the energy budget is set to 8×10^3 (Joule). From this figure, we can see that the achievable total social welfare will reach a plateau at the end. This is because with the number of tasks increasing, each task is always served by its selected mobile user while other unselected tasks cannot increase the achievable total social welfare. Note that when the number of tasks is large enough, the total social welfare is still gradually increasing. It can be explained as follows. Since newly arrived tasks have different deadline requirements, the BS will prefer less delay intensive tasks so as to lower its communication and computation resource demand, resulting in the slow increase of the total social welfare. Moreover, this figure also verifies the superiority of our proposed online mechanism over others, which further demonstrates that our proposed mechanism can always achieve better performance under any combination of offloaded tasks and mobile users.

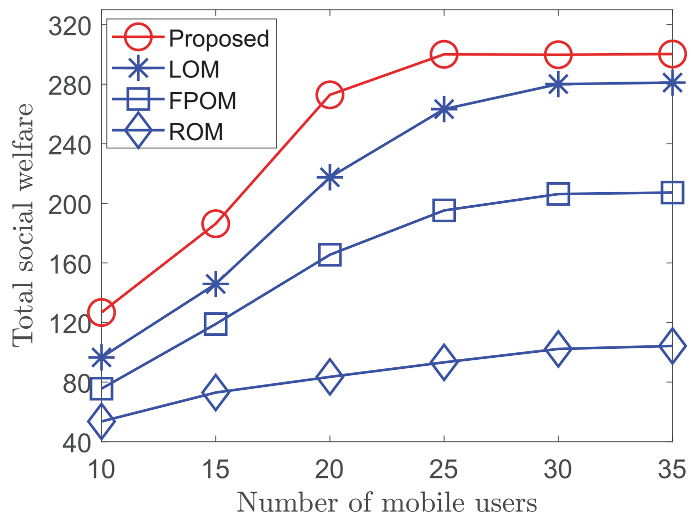

Figure 4.3: Total social welfare versus number of mobile users.

Fig. 4.5 depicts the relationship between the total social welfare and the maximal transmission power at mobile users. It can be observed that the total social welfare by the proposed online mechanism increases first and then keeps stable when the maximal transmission power at mobile users is large enough. This is because, in order to meet the delay requirements of tasks, mobile users have to transmit back the computation results with their maximal but very limited transmission powers. With the increase of the maximal transmission power, more tasks' delay requirements can be satisfied, which leads to an increase in the total social welfare. However, when the maximal transmission power is large enough, e.g., $P_j^{max} = 35$ dBm, the optimal transmission power becomes less than the maximal one. In addition, since all tasks have been successfully offloaded, the total social welfare becomes almost a constant. Furthermore, the total social welfare by the LOM method first increases to a maximum, and then reduces slightly when the maximal transmission power is over 35 dBm. This decreasing trend is intuitive. Since the LOM always works at a maximal upload link transmission power, according to the definition of our objective function, the total social welfare decreases when the total social welfare reaches its maximum at almost 32 dBm. In addition, the total social welfare achieved by the FPOM stays almost unchanged. This is because for the FPROM method, both its upload and download links' transmission powers are fixed so that the maximal transmission power at mobile users has no effects on the total social welfare.

Fig. 4.6 shows the trend of average profit, i.e., average utilities of mobile users and the BS, with

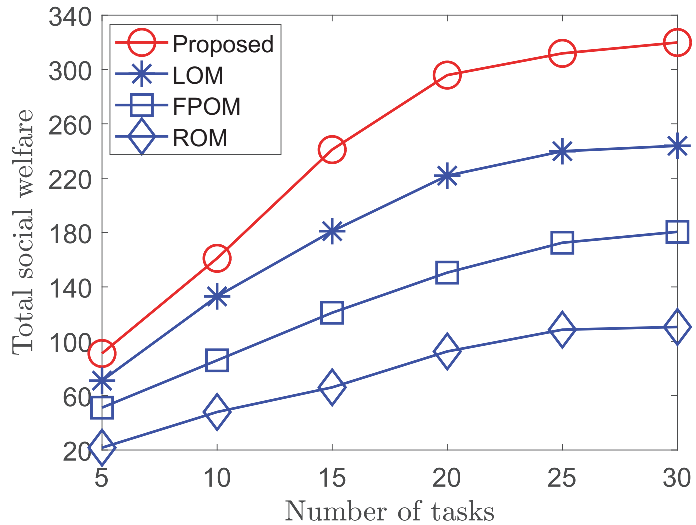

Figure 4.4: Total social welfare versus number of offloaded tasks.

the number of mobile users under different online mechanisms. From this figure, we can observe that both utilities of mobile users and the BS increase with the number of mobile users. This is because more mobile users can accept more offloaded tasks for execution, which increases the benefits. Moreover, since our proposed online mechanism jointly considers the communication and computation resource allocations, it is obvious that our proposed scheme can gain more benefits compared to others. Furthermore, the average utility of mobile users by our proposed online mechanism is no less than zero, which manifests the property of individual rationality holds.

Fig. 4.7 presents the comparisons among different online mechanisms with respect to online to offline ratio (i.e., CR ratio) along system running time. Note that the optimal offline solution is obtained by the brute force method, and to reduce the computation time, we limit both the numbers of mobile users and tasks to be 5. Also, note that by letting $\varphi = 0.1$ and $\beta = 10$, we have the theoretical CR about 0.45, which is the worst-case performance. However, from this figure, we can observe that the actually achievable CR by our proposed mechanism is around 0.75, which is much better than this worst-case CR. Moreover, the actual CR almost stays unchanged along system running time, which shows that the proposed online mechanism is robust to the running time.

Fig. 4.8 reveals the relationship between the total social welfare and the available energy budget at mobile users with different number of mobile user and task pairs. Three cases are under consideration. In case one, the number of tasks is 5 and the number of mobile users is 20; in case two,

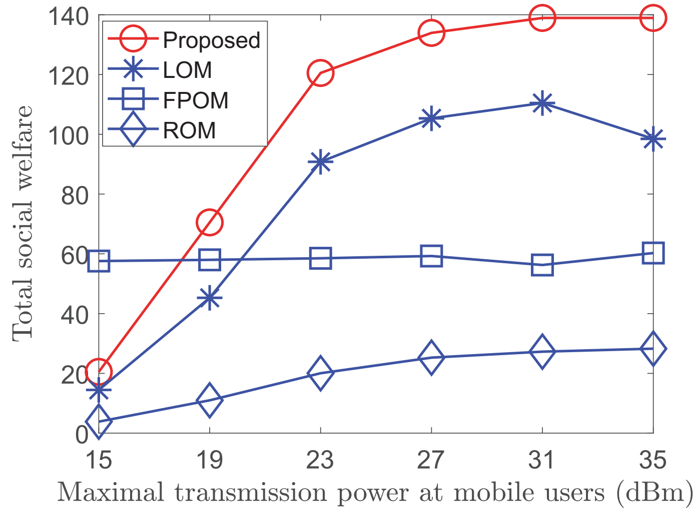

Figure 4.5: Total social welfare versus maximal transmission power at mobile users.

the number of tasks is 20 and the number of mobile users is 5; and in case three, the numbers of both mobile users and tasks are 5. From this figure, it can be seen that the total social welfare first increases and eventually strikes a balance regardless of the amounts of mobile users and tasks. This is because, with the increase of the available energy budget, more mobile users can survive to execute more tasks so that the total social welfare increases. However, since there are only at most five tasks for executions, the total social welfare will not continue increasing when the available budget is large enough. Besides, for case one, the total social welfare experiences a surge in the early increasing stage with the energy budget. This is because when the energy budget increases a little bit, twenty mobile users could be enough to alternatively execute five offloaded tasks. Moreover, the total social welfare under case two, is always larger than that under case three. The reason behind this can be explained as follows. In case two, there is a sufficient number of tasks so that there will always be tasks with less stringent delay tolerance requirements to choose from. Hence, both mobile users and the BS could consume less computation and communication resources for offloaded task execution so that the total social welfare becomes higher. In addition, the total social welfare by case two surpasses that by case one when the available energy budget becomes large enough. This is because compared to the diversity of mobile users, the diversity of tasks will contribute more to the increase of the total social welfare, which can also be corroborated by comparing Fig. 4.3 and Fig. 4.4 in the stable state.

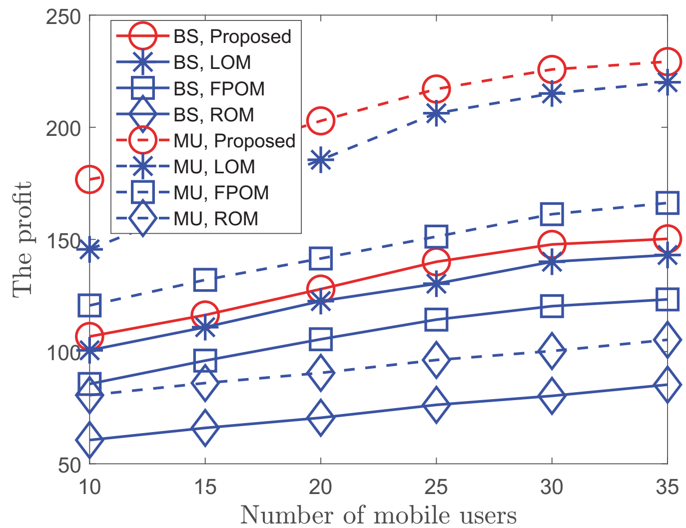

Figure 4.6: The profits versus number of mobile users.

4.4 Summary

In this chapter, a nonlinear online truthful mechanism for task offloading in edge computing systems has been proposed. By considering the facts that the arrival of tasks is dynamic and each mobile user is energy-constrained in practice, we formulate a social welfare maximization problem by jointly considering task offloading decisions, and both computation and communication resource allocation. We convert this time coupled incentive mechanism design problem into several one-shot ones, and solve them by our proposed IRSM framework. Finally, we rationally combine the results of one-shot incentive mechanism to design a nonlinear online incentive mechanism. Theoretical analyses guarantee the properties of IR, IC, and computational efficiency. Numerical results further demonstrate the effectiveness and superiority of our proposed nonlinear online truthful mechanism.

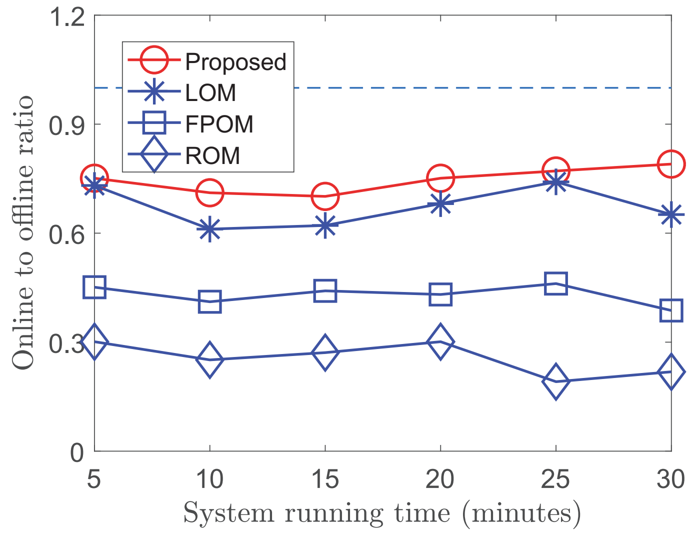

Figure 4.7: Online to offline ratio versus system running time.

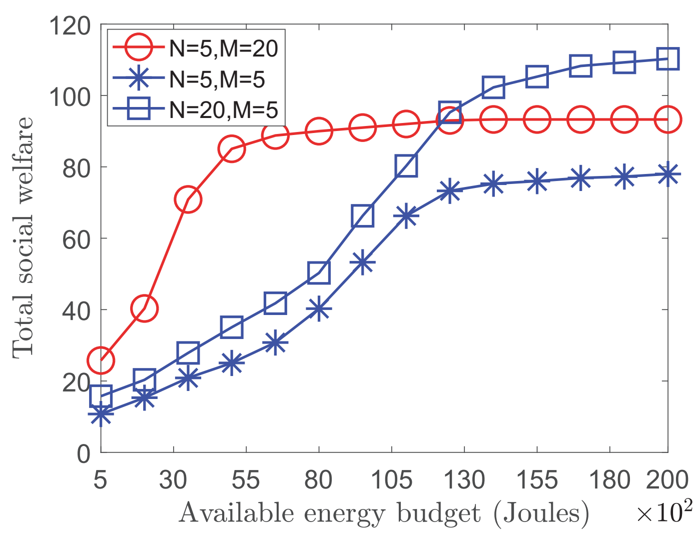

Figure 4.8: Total social welfare versus available energy budget at mobile users.

Chapter 5

Truthful Deep Mechanism Design for Revenue-Maximization in Edge Computing with Budget Constraints

Incentive mechanism design commonly needs to solve very complicated optimization problems. For example, for the objective of revenue maximization, the mechanism design has to jointly optimize resource allocation and pricing rule, which is a multi-dimensional, nonlinear, and long-term optimization problem. In this chapter, collaborative task offloading in EC systems is studied to maximize the net revenue of service provider, where computation requesters can offload tasks to not only the edge server, but also nearby mobile users. By considering the fact that mobile users may not always be willing to provide such computation service because of the consumption of their own energy and resources, an incentive mechanism is designed to provide incentive to mobile users. We aim to design a new incentive mechanism integrating deep learning approach, i.e., truthful deep mechanism design, for collaborative task offloading in an EC system. Our objective is to maximize the net revenue of the service provider while considering economical properties in terms of individual rationality, incentive compatibility, and budget balance. Because of the high computational complexity involved, which prevents the application of traditional mechanism design methods, new

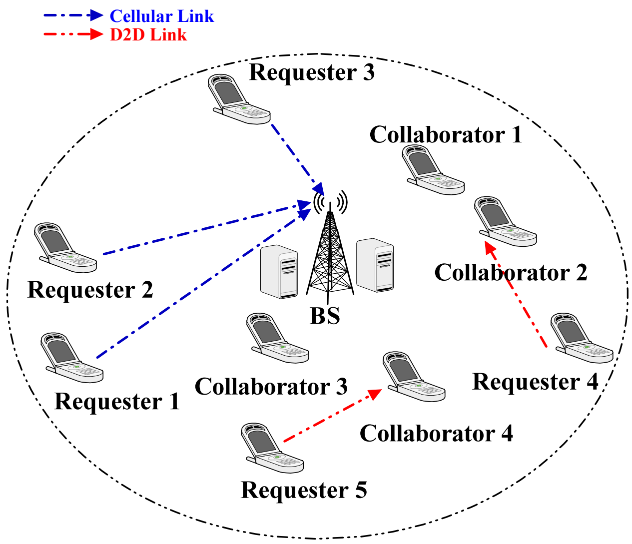

Figure 5.1: Collaborative task offloading in edge computing.

approach based on a multi-task machine learning model is devised to generate results for two inherently interconnected tasks, i.e., collaborator selection and pricing rule, simultaneously. Specifically, our proposed multi-task machine learning model consists of two related well-designed deep neural networks for collaborator selection and pricing rule, respectively. Then, the outcomes of both deep neural networks jointly determine a loss function, which is the optimization objective of our multi-task machine learning model. Finally, this devised multi-task machine learning model is trained under this defined loss function. The numerical results show that the proposed deep truthful mechanism can ensure a convergence to a stable state and can satisfy all required economical properties, including individual rationality, incentive compatibility, and budget balance.

5.1 System model and Problem formulation

In this section, the system model is first introduced. After that we formulate the interactions between the BS and requesters, i.e., mobile users, as an incentive mechanism design problem.

5.1.1 System Description

Similar to [41, 55], we consider an EC system, as shown in Fig. 5.1. In the system, there is a BS equipped with an EC server. Mobile users are classified into two categories. $|\mathcal{N}| = N$ users are called requesters, who would like to offload computation tasks, while the rest of $|\mathcal{M}| = M$ mobile users are collaborators, who are recruited by the BS at the beginning of offloading process and willing to provide computation service if rewards are provided. The computation tasks can come from the applications, such as data compression, face recognition, and virus scan. Each collaborator can serve at most one task at any time because of constraints on its computation capacity and the resulted processing delay. By adopting a similar term in chapter 3, we still use the "task executor" to represent either the collaborator or the BS, and each requester can only offload its task to at most one task executor. In the considered system model, each requester first submits its preference value, requested computation resource, available budget, and available collaborators[1] to the BS, who then assigns task executors to complete all requested tasks. After completion, the task executors send results to the corresponding requesters, and the requesters should make compensation to the corresponding task executors, both of which could be done through D2D links [41] or cellular links. Note that if computational results are not received successfully by the corresponding requester, no reimbursement will be given to the task executor. Therefore, collaborators will try their best to complete the offloading process[2].

Obviously, such interactions between the task executors and the requesters can be modelled as a truthful mechanism design problem, where the requesters are buyers and the task executors are sellers. In this truthful mechanism design, for buyer $i \in \mathcal{N}$, its submitted bid can be denoted as $b_i = \{b_{i,1}, b_{i,2}, \cdots b_{i,M+1}, f_i, B_i\}$, where $b_{i,j}$ is the submitted valuation of buyer i for task executor j, f_i denotes the corresponding requested computation frequency, and B_i denotes the budget of mobile user i. We further define a matrix $\boldsymbol{b} = \{b_1; b_2; \cdots ; b_N\}$, which is composed of all requesters' bids, and denote $\boldsymbol{v}$ as the truthful bids of $\boldsymbol{b}$. As previously stated, it is practical that $\boldsymbol{b}$ are private

[1] Note that the available collaborators nearby requester i can be found by applying the discovery approach [95].

[2] If the collaborator moves out of the communication coverage of requesters just after the start of offloading process, the task will probably not be completed because of transmission interruption, which incurs no rewards to collaborators. Therefore, there are no incentives for collaborators to move out. For the movement of requester, the computation result and the reimbursements can also be received and transmitted through the cellular link even if requesters move out of the cell.

known information to requesters themselves and may vary with the time. Therefore, similar to works in [38, 60, 115], the bid of mobile user i follows a distribution F_i. But, unlike those work, which always assumed the availability of this *a prior* distribution, we consider a more general case where this distribution may be unavailable. Moreover, let l_i be the task size in bit unit of requester i, and the index $M+1$ represents the BS and $\mathcal{M}_1 = \{M+1 \cup \mathcal{M}\}$. Note that requesters can have different values of $b_{i,j}$ to different task executors. For example, a requester with a latency-intensive task may value more to collaborators who have better channel conditions, while the requester with a computation-intensive task may value more to the BS.

Our designed truthful mechanism consists of an allocation rule (i.e., collaborator selection) $g(\boldsymbol{b}) = \{g_{i,j}(\boldsymbol{b})\}_{i\in\mathcal{N},\, j\in\mathcal{M}_1}$ and a pricing or reward policy $p(\boldsymbol{b}) = \{p_i(\boldsymbol{b})\}_{i\in\mathcal{N}}$, where the function $g_{i,j}(\boldsymbol{b})$ specifies winners that can offload tasks, while the function $p_i(\boldsymbol{b})$ stipulates the amount of rewards or reimbursements to task executors. Therefore, the utility of any requester i can be calculated as

$$U_i(\boldsymbol{b}) = \sum_{j\in\mathcal{M}_1} g_{i,j}(\boldsymbol{b}) b_{i,j} - p_i(\boldsymbol{b}). \tag{89}$$

Note that in the bidding process, each requester may strategically misreport its bid in order to maximize its utility. As such, in the proposed truthful mechanism design, the following three properties in this chapter have to be met to prevent misreport and incentivize participation.

Definition 12 (IR)**.** *A truthful mechanism $M^R = (g(\boldsymbol{b}), p(\boldsymbol{b}))$ is individually rational for all requesters, if their utilities are no less than 0, i.e.,*

$$U_i(\boldsymbol{b}) \geq 0, \quad \forall\, i \in \mathcal{N}. \tag{90}$$

Definition 13 (IC)**.** *A truthful mechanism $M^R = (g(\boldsymbol{b}), p(\boldsymbol{b}))$ is incentively compatible if no requester can improve its utility by misreporting its bid, i.e.,*

$$U_i(b_i, \boldsymbol{b}_{-i}) \geq U_i(\hat{b}_i, \boldsymbol{b}_{-i}), \quad \hat{b}_i \in \eta(i),\ \forall\, i \in \mathcal{N}, \tag{91}$$

where $\boldsymbol{b}_{-i} \in \boldsymbol{b}$ is defined as the set of bids from other requesters expect i, and $\eta(i)$ as the set of *potential misreport of requester i.*

Definition 14 (BB)**.** *A truthful mechanism $M^R = (g(\boldsymbol{b}), p(\boldsymbol{b}))$ is budget balanced if no requesters' payments exceed their budgets, i.e.,*

$$p_i(\boldsymbol{b}) \leq B_i,\ \forall\, i \in \mathcal{N}. \tag{92}$$

Besides, since each task executor has limited computation resource capacity, we impose the following constraint on each task executor as

$$\sum_{i \in \mathcal{N}} f_i g_{i,j}(\boldsymbol{b}) \leq F_j,\ \forall\, j \in \mathcal{M}_1, \tag{93}$$

where F_j is the maximal computation resource at executor j. Furthermore, by considering the fact that each mobile user can be offloaded to at most one task executor, and each collaborator can only execute at most one offloaded task, we have the following constraints:

$$\sum_{i \in \mathcal{N}} g_{i,j}(\boldsymbol{b}) \leq 1,\ \forall\, j \in \mathcal{M}, \tag{94}$$

$$\sum_{j \in \mathcal{M}_1} g_{i,j}(\boldsymbol{b}) \leq 1,\ \forall\, i \in \mathcal{N}. \tag{95}$$

Note that constraints (94) and (95) exclude the BS since the BS can accept more than one task resulting from its ample computational capacity.

5.1.2 Problem Formulation

Similar to [37, 38], in this chapter, we aim to derive functions $g(\boldsymbol{b})$ and $p_i(\boldsymbol{b})$ that maximize the expected net revenue, which can be expressed as

$$\mathcal{L}(g(\boldsymbol{b}), p(\boldsymbol{b})) = \mathbb{E}_{\boldsymbol{b} \sim \mathbb{F}}\{\sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{M}_1} (p_i(\boldsymbol{b}) - c_{i,j}) g_{i,j}(\boldsymbol{b})\}, \tag{96}$$

where $\mathbb{F} = F_1 \times F_2 \times \cdots \times F_N$ is the joint distribution of all requesters' biding $\boldsymbol{b}$, $c_{i,j} = \beta_j \times \xi_j l_i f_i^2$ is the processing cost of executor j for requester i, β_j, $j \in \mathcal{M}_1$ is the cost for unit energy consumption at task executor j, and ξ_j is the energy consumption coefficient [99]. In the following discussions, we omit the vector $\boldsymbol{b}$ in both $g(\boldsymbol{b})$ and $p_i(\boldsymbol{b})$ for notational simplification. Then, our revenue maximization mechanism design can be formulated as the following optimization problem:

$$\max_{(g,p)\in\mathcal{C}} \mathcal{L}(g,p) \tag{P1}$$

$$\text{s.t. } (90), (91), (92), (93), (94), \text{and } (95),$$

where $\mathcal{C}$ denotes a class of mechanisms. However, solving such optimization problem (P1) is extremely difficult because 1) even if constraints (92)–(95) are removed from the problem (P1), distributions on requesters' valuations have to be known to solve this problem [37]. However, such distributions are not always available in practice; 2) even though such *a prior* distributions are known in advance, according to [58], existing approaches can only deal with the problem (P1) without considering the constraint (92) and in a small scale case[3]; 3) (g, p) in (P1) are not simple variables, but functions, which make (P1) fall in the scope of functional analysis. To address all aforementioned challenges, in the following section, we will propose a new mechanism design method to solve this complex problem.

5.2 Truthful Deep Mechanism Design

In this section, we first reformulate (P1), and then develop a framework to design a truthful deep mechanism to maximize the net revenue with the consideration of all constraints on IR, IC, BB, and computation capacity.

5.2.1 Problem Reformulation

We first introduce the following metrics to measure the deviation degree from IR, IC, BB, and the constraint (93). For these reasons, we

[3]Small scale means the number of participants in the bidding campaign is small, such as two or three.

- define expected ex-post regret for any requester i as

$$\rho(g, p_i) = \mathbb{E}_{\boldsymbol{b}\sim\mathbb{F}}\{\max_{\hat{b}_i\in\eta(i)} U_i(\hat{b}_i, \boldsymbol{b}_{-i}) - U_i(b_i, \boldsymbol{b}_{-i})\}, \tag{97}$$

  If $\rho(g, p_i) = 0$, requester i cannot gain its utility by misreporting its information, which means the constraint (91) holds.

- define the expected ex-post IR penalty to requester i as

$$\delta(g, p_i) = \mathbb{E}_{\boldsymbol{b}\sim\mathbb{F}}\{\max\{0, -U_i(\boldsymbol{b})\}\}. \tag{98}$$

  If $\delta(g, p_i) = 0$, the utility of requester i is no less than zero, which means the constraint (90) satisfies.

- define the expected BB penalty to requester i as

$$\phi(g, p_i) = \mathbb{E}_{\boldsymbol{b}\sim\mathbb{F}}\{\max\{0, p_i(\boldsymbol{b}) - B_i\}\}. \tag{99}$$

  If $\phi(g, p_i) = 0$, no requesters can have an overpayment, which means the constraint (92) satisfies.

- define the expected computation resource penalty of task executor j as follows

$$\theta_j(g) = \mathbb{E}_{\boldsymbol{b}\sim\mathbb{F}}\{\max\{0, \sum_{i\in\mathcal{N}} f_i g_{i,j}(\boldsymbol{b}) - F_j\}\}. \tag{100}$$

  If $\theta_j(g) = 0$, no executor exceeds its computation capacity, which means the constraint (93) satisfies.

Furthermore, since the joint distribution, i.e., $\mathbb{F}$, of all requesters' biding is unknown to the BS, in order to further simplify the mechanism problem (P1), we use the geometric average to represent (97)-(100) and the objective. To this end, we apply a Q-length sample set $TS = \{\boldsymbol{b}^{(1)}, \cdots, \boldsymbol{b}^{(q)}, \cdots, \boldsymbol{b}^{(Q)}\}$

from the historical data and estimate (96)-(100) based on this set as

$$\hat{\delta}(g,p_i) = \frac{1}{Q}\sum_{q=1}^{Q}\max\{0, -U_i(\boldsymbol{b}^{(q)})\}, \tag{101}$$

$$\hat{\phi}(g,p_i) = \frac{1}{Q}\sum_{q=1}^{Q}\max\{0, p_i(\boldsymbol{b}^{(q)}) - B_i\}, \tag{102}$$

$$\hat{\theta}_j(g) = \frac{1}{Q}\sum_{q=1}^{Q}\max\{0, \sum_{i\in\mathcal{N}} f_i g_{i,j}(\boldsymbol{b}^{(q)}) - F_j\}. \tag{103}$$

Moreover, in order to estimate the constraint (97), another sample set $TS^{(q)}$ independently drawn from the historical data is leveraged as misreport bids for each $\boldsymbol{b}^{(q)}$, and we calculate the maximum utility gain over this sample set $TS^{(q)}$ as

$$\hat{\rho}(g,p_i) = \frac{1}{Q}\sum_{q=1}^{Q}\max_{\hat{b}_i\in TS^{(q)}} U_i(\hat{b}_i, \boldsymbol{b}_{-i}^{(q)}) - U_i(\boldsymbol{b}^{(q)}), \tag{104}$$

where $\boldsymbol{b}_{-i}^{(q)}$ represents the q^{th} element in the sample set $TS^{(q)}$, but not including the bids of requester i. Note that in order to keep in line with the traditional usage in machine learning, we intend to minimize the negative expected net revenue so that the estimation of (96) can be expressed as

$$\hat{\mathcal{NL}}(g,p) = -\frac{1}{Q}\sum_{q=1}^{Q}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{M}_1}(p_i(\boldsymbol{b}^{(q)}) - c_{i,j})g_{i,j}(\boldsymbol{b}^{(q)}). \tag{105}$$

Given these estimates, we can reformulate the original problem (P1) as

$$\min_{(g,p)\in\mathcal{C}} \hat{\mathcal{NL}}(g,p) \tag{P2}$$

$$\begin{aligned}
\text{s.t.}\quad & \hat{C}_1: \ (94),\ (95),\\
& \hat{C}_2: \ \hat{\theta}_j(g) = 0\ \forall j\in\mathcal{M}_1,\\
& \hat{C}_3: \ \hat{\delta}(g,p_i) = 0,\ \forall i\in\mathcal{N},\\
& \hat{C}_4: \ \hat{\rho}(g,p_i) = 0,\ \forall i\in\mathcal{N},\\
& \hat{C}_5: \ \hat{\phi}(g,p_i) = 0,\ \forall i\in\mathcal{N}.
\end{aligned}$$

It is worth noting that g and p in the problem (P2) are optimizing functions. To this end, inspired by the framework of multi-task machine learning, we design a truthful deep mechanism in follows where g and p are both modelled by our designed inter-related neural networks, and are further trained together as a whole neural network. Furthermore, the designed truthful deep mechanism is actually a randomized mechanism, where the allocation scheme g represents the probability that a requester can be allocated to a given task executor.

5.2.2 Design of Allocation Rule and Pricing Policy

Note that in the reformulated problem (P2), our aim is to find suitable allocation rule and pricing policy, i.e., g and p, to minimize the objective while satisfying constraints $\hat{C}_1 - \hat{C}_5$. For this purpose, we model the allocation rule, i.e., g, as a feed-forward neural network which contains L fully-connected hidden layers with sigmoidal activations for each node, as shown in Fig. 5.2. We use the softmax activation function to output a interim vector as $a_{11}, a_{(N+1)1} \cdots z_{11}, z_{N(M+2)}$, and determine the final allocation results by comparing values of those elements in the interim vector. Specifically, in this network, we denote $y_k^{(\ell)}$ as the output of sigmoidal activation function k at the ℓ^{th} hidden layer, where $\ell \in \mathcal{L} = \{1, 2 \cdots, L\}$ and $k \in \mathcal{K}_\ell = \{1, 2, \cdots, J_\ell\}$. J_ℓ is the total number of sigmoidal activation functions at the hidden layer ℓ, and we index each hidden layer's sigmoid functions from top to bottom as $1, 2 \cdots J_\ell$. Let $\boldsymbol{w}_k^{(\ell)}$ be the row weight vector belonging to the k^{th} sigmoid function at hidden layer ℓ. We further use the row vectors $\boldsymbol{w}_{i,j,1}^{(L+1)}$ and $\boldsymbol{w}_{i,j,2}^{(L+1)}$ to represent the weights for output layer $a_{i,j}$ and $z_{i,j}$, respectively, and use $\boldsymbol{w} = \{\boldsymbol{w}_k^{(\ell)}, \boldsymbol{w}_{i,j,1}^{(L+1)}, \boldsymbol{w}_{i,j,2}^{(L+1)}\}$ to denote all vector parameters which are arranged in a row vector. Thus, the outputs of the k^{th} sigmoid function at the first and any hidden layer ℓ can be expressed as

$$d_k^{(1)} = \boldsymbol{w}_k^{(1)} \boldsymbol{I}; \quad y_k^{(1)} = \sigma(d_k^{(1)}), \ \forall\, k \in \mathcal{K}_1, \tag{106}$$

$$d_k^{(\ell)} = \boldsymbol{w}_k^{(\ell)} \boldsymbol{y}^{(\ell-1)}; \quad y_k^{(\ell)} = \sigma(d_k^{(\ell)}), \ \forall\, \ell \in \mathcal{L}; \ k \in \mathcal{K}_\ell, \tag{107}$$

where the column vector $\boldsymbol{I} = \{I_1, I_2, \cdots, I_{N(M+3)}\}^H$ is the input of allocation rule architecture, and the sigmoid function is $\sigma(x) = \frac{1}{1+e^{-x}}$, and $\boldsymbol{y}^{(\ell)} = \{y_1^{(\ell)}, y_2^{(\ell)}, \cdots, y_{J_\ell}^{(\ell)}\}^T$ which is a column

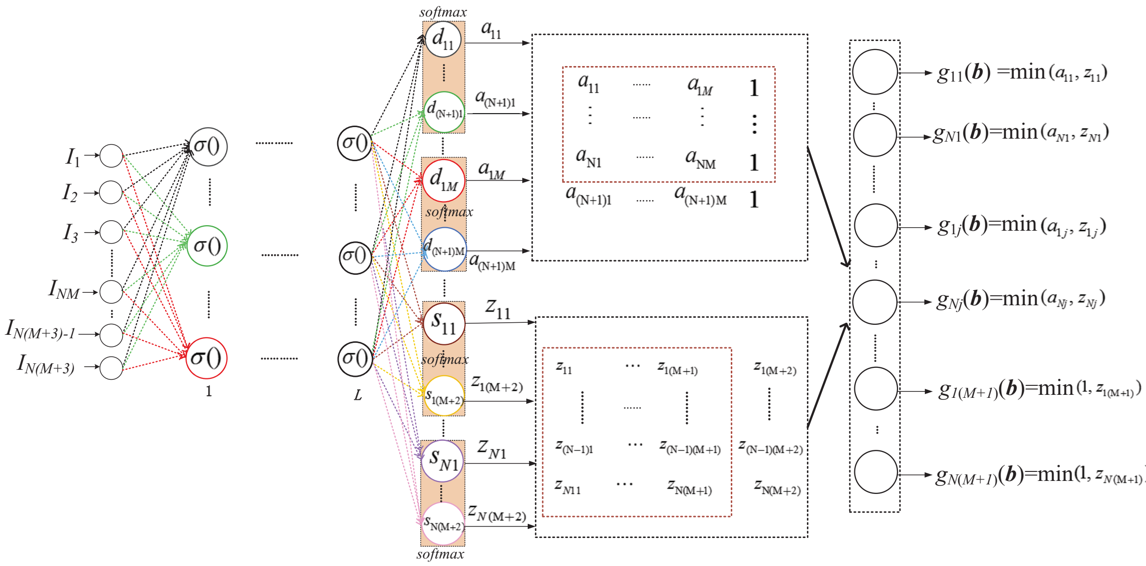

Figure 5.2: Allocation rule architecture g.

vector. The outputs of the last layer in this neural network are

$$d_{i,j}^{(L+1)} = \boldsymbol{w}_{i,j,1}^{(L+1)} \boldsymbol{y}^{(L)},\ \forall i = 1, 2 \cdots, N+1;\ \forall j \in \mathcal{M}, \tag{108}$$

$$s_{i,j}^{(L+1)} = \boldsymbol{w}_{i,j,2}^{(L+1)} \boldsymbol{y}^{(L)},\ \forall i \in \mathcal{N};\ \forall j = 1, 2 \cdots, M+2, \tag{109}$$

$$a_{i,j} = softmax(d_{1,j}^{(L+1)}, \cdots, d_{N+1,j}^{(L+1)}),\ \forall i \in \mathcal{N};\ \forall j \in \mathcal{M}_1,$$

$$z_{i,j} = softmax(s_{i,1}^{(L+1)}, \cdots, s_{i,(M+2)}^{(L+1)}),\ \forall i \in \mathcal{N};\ \forall j \in \mathcal{M}_1,$$

where the softmax function is defined as $softmax(x_1, \cdots, x_n) = \frac{e^{x_i}}{\sum_{k=1}^{n} e^{x_k}}$. Note that in order to indicate the cases where no requester chooses collaborator j or the BS, so that requester i fails to offload its task, $d_{N+1,j}^{(L+1)}$ and $s_{i,M+2}^{(L+1)}$ are used as the reductant inputs, respectively. Let $\mathcal{A}$ and $\mathcal{Z}$ be matrixes of all $a_{i,j}$ and $z_{i,j}$, respectively. Furthermore, by considering the fact that the BS can accept more than one task, we add one identity column vector in the last column of matrix $\mathcal{A}$. Finally, the allocation result $g_{i,j}^{\boldsymbol{w}}(\boldsymbol{b})$ can be obtained by element-wise minimization of matrix $\mathcal{A}$ and matrix $\mathcal{Z}$, i.e., $g_{i,j}^{\boldsymbol{w}}(\boldsymbol{b}) = \min\{a_{i,j}, z_{i,j}\}$.

Similarly, the pricing policy is also modeled by using a feed-forward neural network with $T = |\mathcal{T}|$ fully-connected hidden layers and a fully-connected output layer, as shown in Fig. 5.3. We use

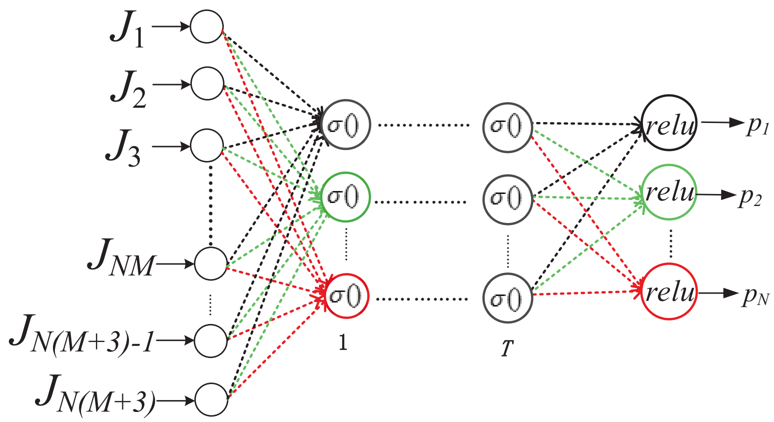

Figure 5.3: Pricing policy architecture p.

$\boldsymbol{w}_k^{'(t)}$ to denote the weight vector of node k at layer t in the price architecture and the output of node k at the hidden layer t is $c_k^{(t)}$. Let all outputs at the hidden layer t be a column vector, denoted as $\boldsymbol{c}^{(t)} = \{c_1^{(t)}, c_2^{(t)}, \cdots, c_{J_t'}^{(t)}\}^H$, where J_t' is the total number of sigmoidal activation functions at the hidden layer t. Also, let $\mathcal{K}_t = \{1, 2, \cdots, J_t'\}$. Likewise, the outputs of the k^{th} sigmoid function at the first layer and any hidden layer t can be respectively written as

$$h_k^{(1)} = \boldsymbol{w}_k^{'(1)}\boldsymbol{J}; \quad c_k^{(1)} = \sigma(h_k^{(1)}), \ \forall\, k = 1, 2, \cdots, J_1', \tag{110}$$

$$h_k^{(t)} = \boldsymbol{w}_k^{'(t)}\boldsymbol{c}^{(t-1)}; \ c_k^{(t)} = \sigma(h_k^{(t)}), \ \forall\, t \in \mathcal{T}; \forall\, k \in \mathcal{K}_t. \tag{111}$$

where the column vector $\boldsymbol{J} = \{J_1, J_2, \cdots, J_{N(M+3)}\}^H$ is the input of pricing policy architecture. The outputs of the last layer in this neural network are

$$h_i^{(T+1)} = \boldsymbol{w}_i^{'(T+1)}\boldsymbol{c}^{(T)}; \ p_i^{\boldsymbol{w}'}(\boldsymbol{b}) = relu(h_i^{(T+1)}), \forall\, i \in \mathcal{N}, \tag{112}$$

where $relu(x) = \max\{x, 0\}$ ensures that payments are non-negative. We use $\boldsymbol{w}' = \{\boldsymbol{w}_k^{'(t)}, t \in \mathcal{T} \cup (T+1); k \in \mathcal{K}_t\}$ to denote all vector parameters which are also arranged in a row vector, and let $p_i^{\boldsymbol{w}'}(\boldsymbol{b})$ represent the payment made by requester i given the pricing policy architecture parameters $\boldsymbol{w}'$ and the biding $\boldsymbol{b}$.

Lemma 18. *According to the designed allocation rule, the allocation result, i.e., $g_{i,j}(b)$, satisfies both constraints* (94) *and* (95).

Proof. Since the allocation rule outputs $a_{i,j}$ and $z_{i,j}$ by softmax function, as in Fig. 5.2, $a_{i,j}$ and $z_{i,j}$ take values between zero and one. For any collaborator j, the softmax function is calculated based on all requesters, which is the column of matrix $\mathcal{A}$. For any requester i, the softmax function is implemented based on each row of matrix $\mathcal{Z}$. Suppose that requester i has the highest probability to offload its task to collaborator j, which means $a_{i,j}$ is the largest value in the jth column in the matrix $\mathcal{A}$. Moreover, collaborator j also coincidentally has the highest chance to accept and execute task from requester i. This means $z_{i,j}$ is the largest value in the ith row in the matrix $\mathcal{Z}$. Therefore, for the other collaborator $j^{'}$, it is impossible to accept the task from requester i as $z_{i,j} > z_{i,j^{'}}$. Even though $a_{i,j^{'}}$ is still a largest one in the $j^{'}$ column, requester i cannot offload task to $j^{'}$ because of $g_{i,j^{'}} = \min\{a_{i,j^{'}}, z_{i,j^{'}}\}$. The same reasons can be applied to another requester $i^{'}$, which cannot offload its task to the same collaborator j. Thus, the final allocation result meets both constraints (94) and (95). This completes the proof. ■

Following the similar procedures as in [116], we can easily prove the following **Lemma**.

Lemma 19. *Let $\boldsymbol{r} = \{r_1, r_2, \cdots, r_Q\}$ be a sample drawn independently from distribution $\mathcal{D}$. Then the following inequality holds with probability of at least $1 - \alpha$.*

$$\mathbb{E}_{\boldsymbol{r}\in\mathcal{D}}\{f(\boldsymbol{r})\} \leq \frac{1}{Q}\sum_{i=1}^{Q} f(r_i) + 2\mathcal{R}_Q(\mathcal{F}) + \sqrt{\frac{\log\frac{1}{\alpha}}{Q}}, \tag{113}$$

where $\mathcal{R}_Q(\mathcal{F}) = \frac{1}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sup_{f(\boldsymbol{r})\in\mathcal{F}} \sum_{r_i\in\boldsymbol{r}} \tau_i f(r_i)\}$, τ_i is a random variable, which is drawn from a uniform distribution on $\{-1,\ 1\}$, and set $\mathcal{F}$ is a class of functions $f(\boldsymbol{r})$.

Theorem 8. *The maximal expected revenue achieved by the proposed machine learning technique is bounded by $\frac{1}{Q}\sum_{q=1}^{Q}\sum_{i\in\mathcal{N}} p_i(b^{(q)}) + 2b_{max}N\sqrt{\frac{2\log(|\mathcal{P}|)}{Q}} + \sqrt{\frac{\log\frac{1}{\alpha}}{Q}}$ with probability of at least $1-\alpha$, where $\mathcal{P}$ is the set of all reward functions.*

Proof. Based on **Lemma** 19, we only need to calculate $\mathcal{R}_Q(\mathcal{F})$. Let $f(\boldsymbol{b}) = \sum_{i\in\mathcal{N}} p_i(\boldsymbol{b})$, and $p_i^{'}(\boldsymbol{b})$

is another payment method from the set $\mathcal{P}'$, such that $\max_{\boldsymbol{b}} \sum_{i\in\mathcal{N}} |p_i(\boldsymbol{b}) - p_i'(\boldsymbol{b})| \leq \epsilon$. We have

$$
\begin{aligned}
&\mathcal{R}_Q(\mathcal{P}) = \frac{1}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sup_{p(\boldsymbol{b})}\sum_{q=1}^{Q}\tau_q\sum_{i\in\mathcal{N}}p_i(\boldsymbol{b}^{(q)})\}\\
&= \frac{1}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sup_{p(\boldsymbol{b})}\sum_{q=1}^{Q}\tau_q\sum_{i\in\mathcal{N}}p_i'(\boldsymbol{b}^{(q)})\} + \frac{1}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sup_{p(\boldsymbol{b})}\sum_{q=1}^{Q}\tau_q\sum_{i\in\mathcal{N}}|p_i(\boldsymbol{b}^{(q)}) - p_i'(\boldsymbol{b}^{(q)})|\}\\
&\leq \frac{1}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sup_{p(\boldsymbol{b})}\sum_{q=1}^{Q}\tau_q\sum_{i\in\mathcal{N}}p_i'(\boldsymbol{b})\} + \frac{\epsilon}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sum_{q=1}^{Q}\tau_q\}\\
&\leq \sqrt{\sum_{q=1}^{Q}(\sum_{i\in\mathcal{N}}p_i'(\boldsymbol{b}^{(q)}))^2}\frac{\sqrt{2\log(|\mathcal{P}'|)}}{Q}\\
&\leq Nb_{max}\sqrt{\frac{2\log(|\mathcal{P}'|)}{Q}},
\end{aligned}
$$

where b_{max} is the maximum submitted valuation from the distribution $\mathbb{F}$, and the last inequality holds because

$$
\sqrt{\sum_{q=1}^{Q}(\sum_{i\in\mathcal{N}}p_i'(\boldsymbol{b}^{(q)}))^2} \leq \sqrt{\sum_{q=1}^{Q}(Nb_{max})^2} = \sqrt{Q}Nb_{max}.
$$

This completes the proof. ■

Theorem 9. *The expected average ex-post regret, i.e., $\rho^A = \frac{1}{N}\sum_{i\in\mathcal{N}}\rho(g, p_i)$ achieved by the proposed machine learning technique is bounded by $\frac{1}{N}\sum_{i\in\mathcal{N}}\hat{\rho}(g, p_i) + 2b_{max}\sqrt{\frac{2\log(K(\mathcal{C}, \frac{\epsilon}{2}))}{Q}} + \frac{1}{N}\sqrt{\frac{\log\frac{1}{\alpha}}{Q}}$ with probability of at least $1-\alpha$, where $K(\mathcal{C}, \frac{\epsilon}{2}))$ is the minimum covering number for the truthful mechanism set $\mathcal{C}$ by a ball with radius $\frac{\epsilon}{2}$.*

Proof. At beginning, we give some definitions as follows. Let $\mathcal{U}_i$ be the set of all possible utility functions for mobile user i, i.e., $U_i(\boldsymbol{b})$, and $\mathcal{U}$ is defined as a set of all utility functions, i.e., $\{U_i(\boldsymbol{b}),\ i \in \mathcal{N}\}$. Furthermore, Define a new function $t_i(\boldsymbol{b})$ as

$$
t_i(\boldsymbol{b}) = \max_{\hat{b}_i\in\eta(i)} U_i(\hat{b}_i, \boldsymbol{b}_{-i}) - U_i(b_i, \boldsymbol{b}_{-i}).
$$

Let $\mathcal{T}_i$ be the set of the function $t_i(\boldsymbol{b})$ with different $U_i(b) \in \mathcal{U}_i$, and $\mathcal{T} = \{\mathcal{T}_i,\ i \in \mathcal{N}\}$. Moreover, we define a spatial distance between two different utility function $U_i(\boldsymbol{b})$ and $U_i^{'}(\boldsymbol{b})$ as $\max_{\boldsymbol{b},\boldsymbol{b}'} |U_i(\boldsymbol{b}) - U_i^{'}(\boldsymbol{b})|$, and a distance between $\mathcal{U}$ and $\mathcal{U}^{'}$ as $\max_{\boldsymbol{b},\boldsymbol{b}'} \sum_{i\in\mathcal{N}} |U_i(\boldsymbol{b}) - U_i^{'}(\boldsymbol{b})|$. We then define $K(U_i(\boldsymbol{b}), \epsilon)$, $K(\mathcal{U}, \epsilon)$, $K(T_i(\boldsymbol{b}), \epsilon)$, and $K(\mathcal{T}, \epsilon)$ be the minimum number of balls with radius ϵ to cover the set $U_i(\boldsymbol{b})$, $\mathcal{U}$, $T_i(\boldsymbol{b})$, and $\mathcal{T}$ under the correspondingly defined distances, respectively. In addition, the spatial distance between two mechanisms $(g_{i,j}(\boldsymbol{b}),\, p_i(\boldsymbol{b})), (g_{i,j}^{'}(\boldsymbol{b}),\, p_i^{'}(\boldsymbol{b})) \in \mathcal{C}$ is defined as $\max_{\boldsymbol{b}} \sum_{i\in\mathcal{N},\, j\in\mathcal{M}_1} |g_{i,j}(\boldsymbol{b}) - g_{i,j}^{'}(\boldsymbol{b})| + \sum_{i\in\mathcal{N}} |p_i(\boldsymbol{b}) - p_i^{'}(\boldsymbol{b})|$, and $K(\mathcal{C}, \epsilon)$ is the minimum number of balls with radius ϵ to cover the mechanism set $\mathcal{C}$ under this distance. Let $t(\boldsymbol{b}) = \sum_{i\in\mathcal{N}} t_i(\boldsymbol{b})$, and the set $\overline{\mathcal{T}} = \{t(\boldsymbol{b}),\ \forall\, (t_1(\boldsymbol{b}), t_2(\boldsymbol{b}), \cdots, t_N(\boldsymbol{b})) \in \mathcal{T}\}$. Similar to **Theorem** 8, we assume that there is another $t^{'}(\boldsymbol{b}) \in \overline{\mathcal{T}}^{'}$ where $|\overline{\mathcal{T}}^{'}| \le K(\overline{\mathcal{T}}, \epsilon)$, which means $\max_{\boldsymbol{b}} |t(\boldsymbol{b}) - t^{'}(\boldsymbol{b})| \le \epsilon$ holds. Then, we have

$$
\begin{aligned}
&\mathcal{R}_Q(\overline{\mathcal{T}}) = \frac{1}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sup_{t_i(\boldsymbol{b})} \sum_{q=1}^{Q} \tau_q \sum_{i\in\mathcal{N}} t_i(\boldsymbol{b}^{(q)})\} \\
&= \frac{1}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sup_{t_i(\boldsymbol{b})} \sum_{q=1}^{Q} \tau_q \sum_{i\in\mathcal{N}} t^{'}(\boldsymbol{b}^{(q)})\} + \frac{1}{Q}\mathbb{E}_{\boldsymbol{\tau}}\{\sup_{t(\boldsymbol{b})} \sum_{q=1}^{Q} \tau_q \sum_{i\in\mathcal{N}} |t_i(\boldsymbol{b}^{(q)}) - t_i^{'}(\boldsymbol{b}^{(q)})|\} \\
&\le N b_{max} \sqrt{\frac{2\log(K(\overline{\mathcal{T}}, \epsilon))}{Q}}.
\end{aligned} \tag{114}
$$

From (114), we only need to prove $K(\overline{\mathcal{T}}, \epsilon) \le K(\mathcal{C}, \frac{\epsilon}{2})$. In order to prove this inequality, we carry out the following three steps.

- We suppose that there exists an utility function set $\mathcal{U}_i^{'}$ with covering number at most $K(\mathcal{U}_i, \frac{\epsilon}{2})$, such that $\max_{\boldsymbol{b}} |U_i(\boldsymbol{b}) - U_i^{'}(\boldsymbol{b})| \le \frac{\epsilon}{2}$. Then, for any $\boldsymbol{b}$, we have

$$
\begin{aligned}
&|\max_{\bar{b}_i}(U_i(\bar{b}_i, \boldsymbol{b}_{-i}) - U_i(b_i, \boldsymbol{b}_{-i})) - \max_{b_i^{'}}(U_i^{'}(b_i^{'}, \boldsymbol{b}_{-i}) - U_i^{'}(b_i, \boldsymbol{b}_{-i}))| \\
&\le |\max_{\bar{b}_i}(U_i(\bar{b}_i, \boldsymbol{b}_{-i}) - \max_{b_i^{'}}(U_i^{'}(b_i^{'}, \boldsymbol{b}_{-i}) + U_i^{'}(b_i, \boldsymbol{b}_{-i})) - U_i(b_i, \boldsymbol{b}_{-i}))| \\
&\le |\max_{\bar{b}_i}(U_i(\bar{b}_i, \boldsymbol{b}_{-i}) - \max_{b_i^{'}}(U_i^{'}(b_i^{'}, \boldsymbol{b}_{-i})| + |U_i^{'}(b_i, \boldsymbol{b}_{-i})) - U_i(b_i, \boldsymbol{b}_{-i}))| \\
&\le |\max_{\bar{b}_i}(U_i(\bar{b}_i, \boldsymbol{b}_{-i}) - \max_{b_i^{'}}(U_i^{'}(b_i^{'}, \boldsymbol{b}_{-i})| + \frac{\epsilon}{2}.
\end{aligned}
$$

Furthermore, let $\overline{b}_i^* = \arg\max_{\overline{b}_i} U_i(\overline{b}_i, \boldsymbol{b}_{-i})$ and $b_i^{'*} = \arg\max_{b_i^{'}} U_i(b_i^{'}, \boldsymbol{b}_{-i})$. Then, we have

$$\max_{\overline{b}_i} U_i(\overline{b}_i, \boldsymbol{b}_{-i}) = U_i(\overline{b}_i^*, \boldsymbol{b}_{-i}) \leq U_i^{'}(\overline{b}_i^*, \boldsymbol{b}_{-i}) + \frac{\epsilon}{2} \leq$$
$$U_i^{'}(b_i^{'*}, \boldsymbol{b}_{-i}) + \frac{\epsilon}{2} = \max_{b_i^{'}} U_i(b_i^{'}, \boldsymbol{b}_{-i}) + \frac{\epsilon}{2},$$

$$\max_{b_i^{'}} U_i^{'}(b_i^{'}, \boldsymbol{b}_{-i}) = U_i^{'}(b_i^{'*}, \boldsymbol{b}_{-i}) \leq U_i(b_i^{'*}, \boldsymbol{b}_{-i}) + \frac{\epsilon}{2} \leq$$
$$U_i(b_i^*, \boldsymbol{b}_{-i}) + \frac{\epsilon}{2} = \max_{b_i} U_i(b_i, \boldsymbol{b}_{-i}) + \frac{\epsilon}{2}.$$

Therefore, for any $U_i(\boldsymbol{b})$, there always exists $U_i^{'}(\boldsymbol{b})$, which satisfies $|\max_{\overline{b}_i}(U_i(\overline{b}_i, \boldsymbol{b}_{-i}) - U_i(b_i, \boldsymbol{b}_{-i})) - \max_{b_i^{'}}(U_i^{'}(b_i^{'}, \boldsymbol{b}_{-i}) - U_i^{'}(b_i, \boldsymbol{b}_{-i}))| \leq \epsilon$. This means $K(\mathcal{T}_i, \epsilon) \leq K(\mathcal{U}_i, \frac{\epsilon}{2})$ holds.

- Assume that there exists another mechanism set $\hat{\mathcal{C}}$, satisfying $|\hat{\mathcal{C}}| \leq K(\mathcal{C}, \epsilon)$. Therefore, for any $(\hat{g}_{i,j}(\boldsymbol{b}), \hat{p}_i(\boldsymbol{b})) \in \hat{\mathcal{C}}$, we have

$$\max_{\boldsymbol{b}} \sum_{i,j} |g_{i,j}(\boldsymbol{b}) - \hat{g}_{i,j}(\boldsymbol{b})| + \sum_i |p_i(\boldsymbol{b}) - \hat{p}_i(\boldsymbol{b})| \leq \epsilon.$$

Remind that $v_{i,j}$ is the truthful valuation to mobile user i, and for the utility functions $U_i(\boldsymbol{b})$ and $\hat{U}_i(\boldsymbol{b})$, we have

$$|U_i(\boldsymbol{b}) - \hat{U}_i(\boldsymbol{b})| \leq$$
$$|\sum_{i,j} v_{i,j} g_{i,j}(\boldsymbol{b}) - \sum_{i,j} v_{i,j} \hat{g}_{i,j}(\boldsymbol{b})| + |p_i(\boldsymbol{b}) - \hat{p}_i(\boldsymbol{b})| \leq$$
$$||v_i||_\infty \sum_j |g_{i,j}(\boldsymbol{b}) - \hat{g}_{i,j}(\boldsymbol{b})| + |p_i(\boldsymbol{b}) - \hat{p}_i(\boldsymbol{b})| \leq$$
$$\sum_j |g_{i,j}(\boldsymbol{b}) - \hat{g}_{i,j}(\boldsymbol{b})| + |p_i(\boldsymbol{b}) - \hat{p}_i(\boldsymbol{b})|,$$

where $v_i = \{v_{i,1}, v_{i,2}, \cdots, v_{i,M+1}\}$, and $||v_i||_\infty$ is the infinite norm of v_i. For all mobile users, we have

$$\sum_i |U_i(\boldsymbol{b}) - \hat{U}_i(\boldsymbol{b})| \le$$
$$\sum_{i,j} |g_{i,j}(\boldsymbol{b}) - \hat{g}_{i,j}(\boldsymbol{b})| + \sum_i |p_i(\boldsymbol{b}) - \hat{p}_i(\boldsymbol{b})| \le \epsilon.$$

This means we have $K(\mathcal{U}, \epsilon) \le K(\mathcal{C}, \epsilon)$.

- Based on the definition of $K(\mathcal{U}, \epsilon)$, there exists $\hat{\mathcal{U}}$ whose covering number is at most $K(\mathcal{U}, \epsilon)$. Therefore, for any $U_i(\boldsymbol{b}) \in \mathcal{U}$, we have $\sum\limits_i |U_i(\boldsymbol{b}) - \hat{U}_i(\boldsymbol{b})| \le \epsilon$. Furthermore, we have $|\sum\limits_i U_i(\boldsymbol{b}) - \sum\limits_i \hat{U}_i(\boldsymbol{b})| \le \epsilon$, which means that $K(\overline{\mathcal{T}}, \epsilon) \le K(\mathcal{T}, \epsilon)$. Moreover, following previous two steps, we have $K(\overline{\mathcal{T}}, \epsilon) \le K(\mathcal{U}, \frac{\epsilon}{2}) \le K(\mathcal{C}, \frac{\epsilon}{2})$.

By applying **Lemma** 19, and taking into consideration of $t(\boldsymbol{b})$, we have

$$\rho^A \le \frac{1}{N} \sum_{i \in \mathcal{N}} \hat{\rho}(g, p_i) + 2b_{max} \sqrt{\frac{2\log(K(\mathcal{C}, \frac{\epsilon}{2}))}{Q}} + \frac{1}{N} \sqrt{\frac{\log \frac{1}{\alpha}}{Q}}.$$

This completes the proof. ■

Note that from **Theorems** 8 and 9, when the number of samples, i.e., Q, is large enough, the expected revenue and the expected average ex-post regret equal the empirical revenue and the empirical average regret.

5.2.3 The Design of Training Method

After we have designed the architectures for both the allocation rule and pricing policy, we can integrate them into a single one, as shown in Fig. 5.4. Since the allocation rule and pricing policy architectures respectively output allocation rule, i.e., $g(\boldsymbol{b})$, and pricing policy, i.e., $p(\boldsymbol{b})$, the utility of each requester can be obtained based on (89). Then, IR, IC, BB, and the constraint (93) can be formulated as constraints in the problem (P1). Note that Fig. 5.4 specifies a multi-task machine learning model resulting from the facts that 1) the determinations of the allocation and pricing rules are two related tasks; 2) the first few layers are shared by two architectures, and these two

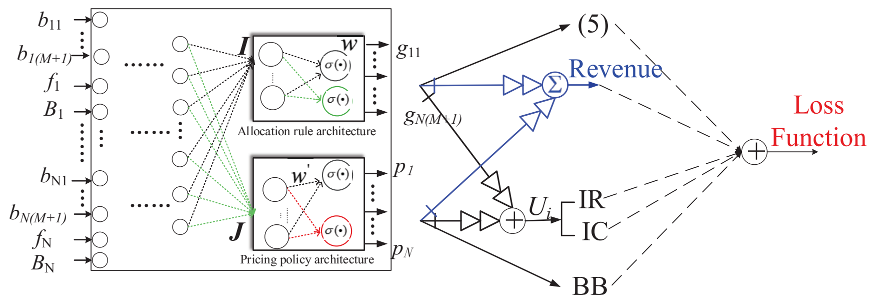

Figure 5.4: The whole multi-task machine learning model of our proposed method

architectures should be trained simultaneously to maximize the net revenue under our defined loss function below.

According to Fig. 5.4 and the problem (P2), we can formulate a new optimization problem (P3) in which the optimization variables are the parameters involved in both allocation rule and pricing policy architectures, i.e., $\boldsymbol{w}$ and $\boldsymbol{w}^{'}$, as follows:

$$\min_{\boldsymbol{w},\boldsymbol{w}^{'}} \mathcal{N}\hat{\mathcal{L}}(g^{\boldsymbol{w}}, p^{\boldsymbol{w}^{'}}) \tag{P3}$$

$$\text{s.t.} \quad \hat{\rho}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}}) = 0, \forall i \in \mathcal{N}, \tag{115}$$

$$\hat{\delta}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}}) = 0, \ \forall i \in \mathcal{N}, \tag{116}$$

$$\hat{\phi}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}}) = 0, \ \forall i \in \mathcal{N}, \tag{117}$$

$$\hat{\theta}_j(g^{\boldsymbol{w}}) = 0, \ \forall j \in \mathcal{M}_1. \tag{118}$$

We solve this training problem (P3) by using the augmented Lagrangian method [117]. Specifically, we first construct a Lagrangian function, i.e., loss function, and then add four quadratic penalty terms in order not to violate constraints (115), (116), (117), and (118). Thus, we have

$$
\begin{aligned}
&Lag(\boldsymbol{w}, \boldsymbol{w}^{'}, \boldsymbol{\lambda}, \kappa) = \mathcal{N}\hat{\mathcal{L}}(g^{\boldsymbol{w}}, p^{\boldsymbol{w}^{'}}) + \sum_{i\in\mathcal{N}}(\lambda_{1,i}\hat{\delta}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}}) \\
&+ \lambda_{2,i}\hat{\phi}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}}) + \lambda_{3,i}\hat{\rho}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}})) + \sum_{j\in\mathcal{M}_1}\lambda_j\hat{\theta}_j(g^{\boldsymbol{w}}) + \\
&\frac{\kappa}{2}(\sum_{i\in\mathcal{N}}(\hat{\delta}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}})^2 + \hat{\phi}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}})^2 + \hat{\rho}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}})^2) + \sum_{j\in\mathcal{M}_1}\hat{\theta}_j(g^{\boldsymbol{w}})^2),
\end{aligned}
$$

where $\boldsymbol{\lambda} = \{\lambda_{1,1}, \cdots, \lambda_{1,N}, \lambda_{2,1}, \cdots, \lambda_{2,N}, \lambda_{3,1}, \cdots, \lambda_{3,N}, \lambda_1, \cdots, \lambda_{M+1}\}$ are the Lagrange multipliers associated with constraints (115), (116), (117), and (118), and $\kappa > 0$ is the penalty parameter that controls the weight for violating constraints (115), (116), (117), and (118). We then perform the following updates in each iteration s:

$$
(\boldsymbol{w}^{(s+1)}, \boldsymbol{w}^{'(s+1)}) = \arg\min_{\boldsymbol{w},\boldsymbol{w}^{'}} Lag(\boldsymbol{w}, \boldsymbol{w}^{'}, \boldsymbol{\lambda}, \kappa) \tag{119}
$$

$$
\begin{aligned}
\lambda_{1,i}^{(s+1)} &= \lambda_{1,i}^{(s)} + \kappa\hat{\delta}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}}), \\
\lambda_{2,i}^{(s+1)} &= \lambda_{2,i}^{(s)} + \kappa\hat{\phi}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}}), \\
\lambda_{3,i}^{(s+1)} &= \lambda_{3,i}^{(s)} + \kappa\hat{\rho}(g^{\boldsymbol{w}}, p_i^{\boldsymbol{w}^{'}}), \\
\lambda_{j}^{(s+1)} &= \lambda_{j}^{(s)} + \kappa\hat{\theta}_j(g^{\boldsymbol{w}}),
\end{aligned} \tag{120}
$$

where the mini-batch stochastic subgradient descent is applied to solve the problem (119). For clarity, the training algorithm for allocation and pricing architectures is listed in **Algorithm** 10.

Next, we evaluate the computational complexity of the training method by using the same metric as in [92,93]. We use one of neural networks as an example, which has N_I and N_o numbers of input and output nodes, respectively, and D number of hidden layers with N_ℓ nodes for each layer. Then, the computation cost for mini-batch gradient descent algorithm is $O(N_I N_\ell + N_\ell^2 D + N_\ell N_o)$. In our case, the total computation cost is $O((3NMN_\ell + 3NN_\ell + N_\ell M + N_\ell^2(L+T))N_B \times N_\lambda \times \left\lfloor \frac{N_D}{N_B} \right\rfloor)$, where the symbol $\lfloor x \rfloor$ denotes the flooring of real number x, N_λ is the maximum iteration number of λ, and N_D and N_B are the total training data and min-batch sizes, respectively. As the algorithm needs the memory space to store the gradients of the weights in the back propagation process [118], the total memory cost is $O((L+T)\times N_\ell)$. Moreover, since the BS has strong computation capacity in the mobile edge computing system, the training algorithm can be done at the BS [119].

Algorithm 10: Training Algorithm.

```
Initialization;
w^(0) = w_0, w'^(0) = w'_0;
λ^(0)_{1,i} = λ_1, λ^(0)_{2,i} = λ_2, λ^(0)_{3,i} = λ_3, ∀ i ∈ N, λ^(0)_j = λ_4, ∀ j ∈ M_1;
κ = κ_0, n = 0;
while n ≤ N_κ do
    s = 0;
    while s ≤ N_λ do
        k = 1 ;
        while k ≤ N_k do
            Obtain w^(s+1) and w'^(s+1) by optimizing (119) with N_k numbers of
              mini-batch;
        Calculate δ̂(g^w, p_i^{w'}), φ̂(g^w, p_i^{w'}), ρ̂(g^w, p_i^{w'}), and θ̂_j(g^w) by using all test data,
          and update λ^(s)_{1,i}, λ^(s)_{2,i}, λ^(s)_{3,i}, λ^(s)_j by (120) ;
    Update κ with N_κ numbers totally by κ = κ + 0.5/n
Output;
w, w', and NL̂ ;
```

5.3 Numerical Results

In this section, we present simulation results[4] to demonstrate the effectiveness of our proposed mechanism in EC collaborative offloading. We use Tensorflow for implementing the deep learning algorithm and assume that the number of requesters $N = 5$ and the number of collaborators $M = 4$. In our simulation, both allocation rule and pricing policy architectures include 2 hidden layers with 8 nodes each. The batch size in the mini-batch stochastic gradient algorithm is set to 32 for training the allocation rule and pricing policy architectures. In each epoch, we first train the proposed mechanism on the training set and evaluate the revenue on the test set, both of which contain 64000 training samples. In addition, the architectures will keep training with the training data for ten epochs until it converges. The processes of updating $\lambda_{1,i}$, $\lambda_{2,i}$, $\lambda_{3,i}$, and λ_j run every 50

[4]Since no practical data are available, similar to [26,47], we generate the training and test data from the three uniform distributions (i.e., two discrete uniform distributions and one continuous distribution).

iterations with the update of the architectures. κ is updated as $\kappa = \kappa + \frac{0.5}{n}$, where n represents the number of times for updating κ, and κ is updated every 1000 iterations of updating networks. The initial value of κ is 0.5 for both DU I and CU distributions, which will be introduced later, while for DU II distribution, the initial value of κ is set to 1.0 for the purpose of fast convergence. The energy consumption coefficient, i.e., ξ_j, at task executor j is set to 10^{-26} [55], the size of offloading tasks, i.e., l_i, are randomly generated between 10 to 30 MB, and the unit energy cost, i.e., $\beta_j = 0.1$ [55]. Besides, the maximal computation capacities at collaborators and the BS are 2 GHz and 10 GHz, respectively, and the requested computation frequency is randomly produced between [0.4 GHz, 1.44 GHz] [57]. Moreover, define $\overline{\rho}$, $\overline{\delta}$, and $\overline{\phi}$ as the average values of $\hat{\rho}_i$, $\hat{\delta}_i$, and $\hat{\phi}_i$ across all requesters on the test set. Note that $\overline{\rho}$, $\overline{\phi}$, $\overline{\delta}$ and $\overline{\theta}$ represent the indicators of IC, IR, BB, and the constraint (93), respectively, which can be mathematically expressed as.

$$\overline{\rho} = \frac{1}{N}\sum_{i=1}^{N}\hat{\rho}(g, p_i), \quad \overline{\delta} = \frac{1}{N}\sum_{i=1}^{N}\hat{\delta}(g, p_i),$$
$$\overline{\phi} = \frac{1}{N}\sum_{i=1}^{N}\hat{\phi}(g, p_i), \quad \overline{\theta} = \frac{1}{M+1}\sum_{j=1}^{M+1}\hat{\theta}_j.$$

The values of Lagrangian function, revenue, $\overline{\rho}$, $\overline{\phi}$, $\overline{\delta}$ and $\overline{\theta}$ are recorded every batch in the mini-batch stochastic gradient during the training processes. In order to avoid the issue of overfitting in the training process, all results in the following figures are output on the test set after each training epoch. In addition, to study the effect of requesters' different budgets on the revenue, we generate budgets based on two different forms where the first form is a uniform distribution that contains the maximal valuation of each requester, while the other one is also a uniform distribution that any values from this distribution is higher than the maximal valuation of each requester. Furthermore, for better illustrating the generalization of our designed mechanism, we generate requesters' valuations from the following distributions. Note that our proposed mechanism has no limitation on certain distributions.

- Discrete Uniform I (DU I): valuations of each requester are drawn from the identical uniform distribution over two values $(0.5, 1.0)$.

- Budget I (B I): the budgets of each requester are drawn from an identical uniform distribution over $[0.5, 3]$;
- Budget II (B II): the budgets of each requester are drawn from an identical uniform distribution over $[1, 3]$;

- Discrete Uniform II (DU II): valuations are drawn from the identical uniform distribution over three values $(0.5, 1.0, 1.5)$.
  - Budget I (B I): the budgets of each requester are drawn from the identical uniform distribution over $[0.5, 5]$;
  - Budget II (B II): the budgets of each requester are drawn from the identical uniform distribution over $[1.5, 3]$;
- Continuous Uniform (CU): valuations of each requester are drawn from the identical uniform distribution over $[0, 1]$.
  - Budget I (B I): the budgets of each requester are drawn from the identical uniform distribution over $[0, 3]$;
  - Budget II (B II): the budgets of each requester are drawn from the identical uniform distribution over $[1, 3]$;

Fig. 5.5 illustrates the trend of Lagrangian functions through the training processes for different distributions with two budget cases. We can see that the curves of these three distributions decrease, and then gradually stay unchanged after one thousand iterations, which indicates that the networks have been trained to convergence.

Fig. 5.6 evaluates the revenues along the training process. As shown in this figure, all curves reach saturation after a short period of time. Moreover, among a certain budget distribution, DU II achieves the largest revenue. It is because the bidding valuations and the budgets for DU II are larger than other scenarios so that the winning probability of requesters with high valuations increases. Similarly, the revenue of CU with B I is the smallest because both bidding valuations and budgets are the smallest in average so that the reimbursements to collaborators or the BS are relatively low. Interestingly, even if the lower bound of budget distribution is higher than the upper

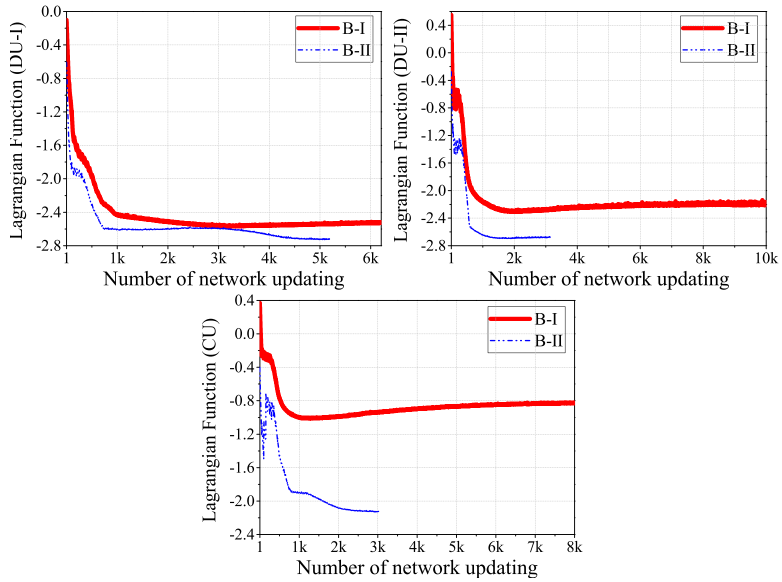

Figure 5.5: The value of Lagrangian function of 3 valuation distributions with two different budgets during network updatings

bound of valuation distributions, i.e., using B II, the revenue of DU I is smaller than that of its counterpart using in B I. It is because by using B II, only two requesters can win and offload their tasks to nearby collaborators, as shown in Table 5.1, so that the total revenue of DU I becomes lowest. Furthermore, for both DU II and CU, when adopting budget distribution B II, the revenue values will be higher than that of its counterpart using B I. This is because requesters with larger budgets are more easier to win and make higher payments in this case.

Fig. 5.7 depicts the trend of average Regret, i.e., $\overline{\rho}$, for the three valuation distributions with different budgets. In this picture, $\overline{\rho}$ are initially large for all cases. This may because requesters would like to get more utilities by misreporting their bidding information. However, for all three valuation distributions, the value of $\overline{\rho}$ decreases rapidly and then converges to almost zero, which means the IC condition is satisfied.

Fig. 5.8 shows the value of BB penalty, i.e., $\overline{\phi}$, with the increase of training process for three valuation distributions with different budgets. It can be shown that initiatively, when the budget distributions which contain the valuation distributions are considered, i.e., using B I, $\overline{\phi}$ are small enough for both DU I and DU II, while the value of $\overline{\phi}$ for CU is a bit higher than others. This is because, at the very beginning, $p^{\omega'}$ in the pricing policy architecture for both DU I and DU II is

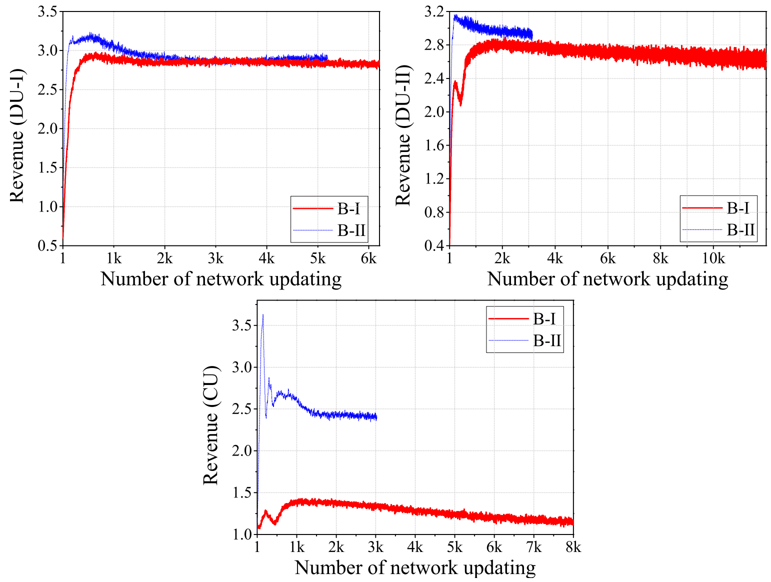

Figure 5.6: The value of Revenue of 3 valuation distributions with two different budgets during network updating

very small, but is relatively larger for CU, which can be verified by initial revenues in Fig. 5.6. Afterwards, $\overline{\phi}$ for those three distributions will eventually reach a plateau before a peak appears at about one thousand iterations. These can be explained as follows. As the ongoing of the training process, all revenues firstly increase so that payments from requesters may violate their budgets. Furthermore, although revenues become stable after about 1000 iterations, the training algorithm is still running to satisfy the BB constraint. As a consequence, a peak of $\overline{\phi}$ will appear shortly for all distributions, and then the values go down and stabilize below 0.01 afterwards, which means the BB condition is satisfied. Note that quite different from the trends by using B I for three valuation distributions, values of $\overline{\phi}$ are all around zero in B II case. This is because the lowest value from budget B II is higher than that of the upper bound of those valuation distributions so that the BB condition is naturally demanded in this case.

Fig. 5.9 demonstrates the change of IR penalty, i.e., $\overline{\delta}$, for three valuation distributions with different budgets. From this figure, we can see that with the increase of iterations, the values of $\overline{\delta}$ for all cases gradually decline, and then tend to almost zero, which means the IR condition is satisfied after training. Note that for all valuation distributions, different budget distributions have no influences on satisfying IR condition. Furthermore, Fig. 5.10 shows the changing trend of computation

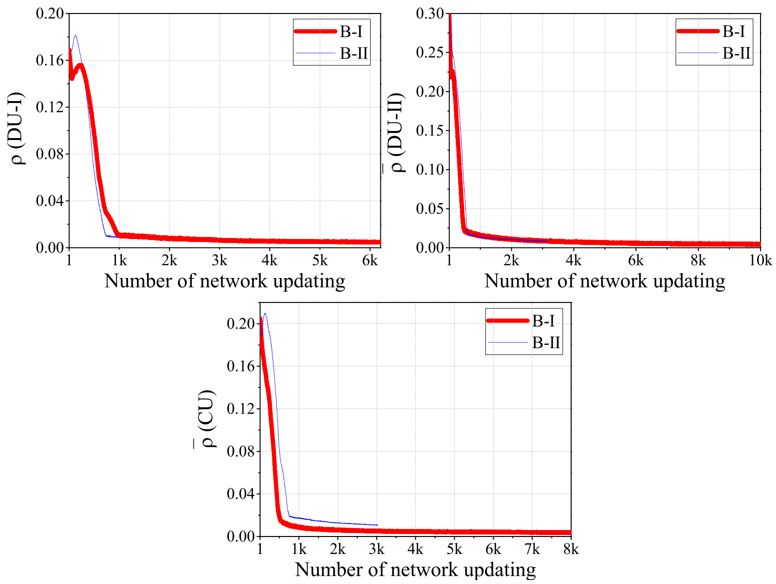

Figure 5.7: The value of $\overline{\rho}$ of 3 valuation distributions with two different budgets during network updatings

resource penalty, i.e., $\overline{\theta}$, for three valuation distributions. It can be seen that allocated computation resources incurs a very small violations on θ_j, which means the constraint (93) holds.

Table 5.1: Offloading Allocation Results by Proposed Incentive Mechanism for Three Valuation Distributions with Two Different Budgets

			DU I					DU II					CU				
			Collaborator				BS	Collaborator				BS	Collaborator				BS
			1	2	3	4		1	2	3	4		1	2	3	4	
B I	Requester	1			√												√
		2							√					√			
		3		√						√						√	
		4	√														
		5									√						
B II	Requester	1						√									√
		2				√											
		3											√				
		4			√				√						√		
		5										√					√
√ indicates that the requester will offload its task to the corresponding BS or collaborator.																	

Table 5.1 lists one of the potential allocation results for three valuation distributions with two budget distributions by using the well trained allocation rule and pricing policy architectures. It is worth noting that in order to get the maximum revenue, requesters may prefer more to offload their

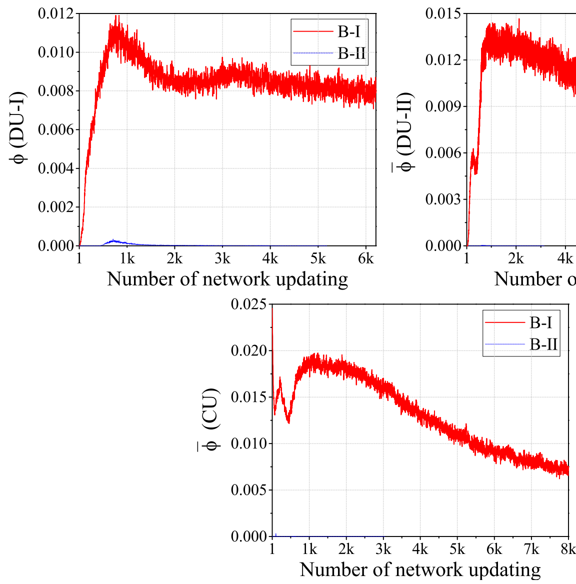

Figure 5.8: The value of $\overline{\phi}$ of 3 valuation distributions with two different budgets during network updatings

tasks to nearby collaborators rather than offload to the BS. This means offloading all tasks to the BS is not always the best choice for requesters, which further shows potential applications of the collaborative offloading architecture.

5.4 Summary

In this chapter, we design a truthful deep mechanism for cooperative task offloading in the edge computing system. Our objective is to maximize the net revenue of the service provider while satisfying IR, IC, BB, and computation resource capacity conditions. The designed mechanism is extremely difficult and no existed methods can be applied to address it effectively. To this end, inspired by multi-task machine learning model, we apply the deep learning technology to achieve our design. Specifically, two specified neural networks architectures are designed to figure out the functions of allocation rule and pricing policy, respectively, and then these neural networks are trained together by the augmented Lagrangian method. Numerical results demonstrate the effectiveness of our designed truthful deep mechanism while only incurring small IR penalty, Regret, BB penalty, and computation resource penalty.

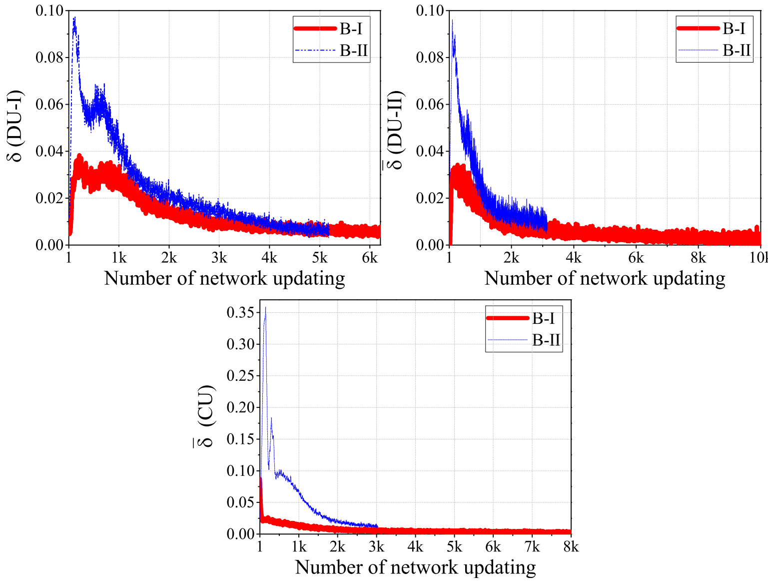

Figure 5.9: The value of $\overline{\delta}$ of 3 valuation distributions with two different budgets during network updatings

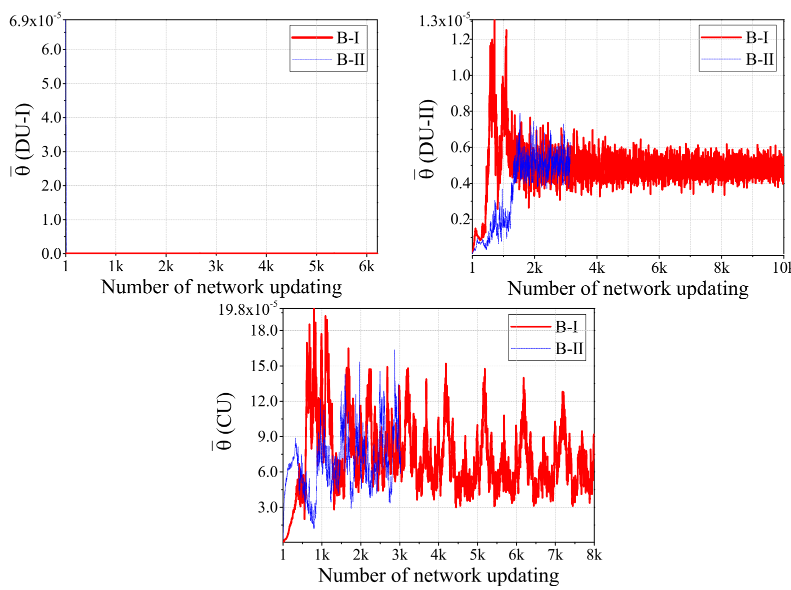

Figure 5.10: The value of $\overline{\theta}$ of 3 valuation distributions with two different budgets during network updatings

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In this thesis, auction-based efficient online incentive mechanism designs in wireless networks have been investigated. Specifically, in Chapter 2, by considering a practical scenario for detecting the availability of parking spaces, an online incentive mechanism is designed by jointly considering the cost budget and the requirement of sensed data of each participant. Specifically, when the task arrives, the platform must make decisions in a sequence to select a specific number of participants to obtain a better competitive ratio. To address this issue, an one-round auction is firstly designed by the utilization of both convex decomposition techniques and VCG pricing rule. Then, an online incentive mechanism is designed by jointly considering designed one-round auction and cost budgets. Moreover, in order to further improve the competitive ratio of the online incentive mechanism, a more efficient online scheme is proposed if more information on the participants is available at the platform. Theoretical and simulation results demonstrate the effectiveness of our proposed online truthful mechanisms.

In Chapter 3, we discuss incentive mechanism design for collaborative task offloading in EC systems. An online incentive mechanism integrating computation and communication resource allocation is proposed. In our system model, upon the arrival of a mobile user who requests task offloading, the BS needs to make a decision right away without knowing any future information on i) whether to accept or reject this task offloading request and ii) if accepted, who to execute the task (the BS itself or nearby mobile users called collaborators). By considering each task's specific requirements in terms of data size, delay, and preference, we formulate a social-welfare-maximization problem, which integrates collaborator selection, communication and computation resource allocation, transmission and computation time scheduling, as well as pricing policy design. To solve this complicated problem, a novel online mechanism is proposed based on the primal-dual optimization framework. Theoretical analyses show that our mechanism can guarantee feasibility, truthfulness, and computational efficiency (competitive ratio of 3). We further use comprehensive simulations to validate our analyses and the properties of our proposed mechanism.

In Chapter 4, we still consider task offloading in EC systems, but tasks are offloaded from the base station to resourceful mobile users. We formulate an incentive mechanism design problem with the consideration of the following three aspects. i) Computation tasks from IoT devices are generated along time, so that offloading decisions at the BS have to be made in an online manner without knowing information on possible future tasks. ii) It is well recognized that both communication and computation resources are limited at both the edge server and the mobile users. Therefore, in order to rationally utilize these limited resources and serve more tasks with their delay requirements, jointly optimizing communication resources, including transmission power and bandwidth for both upload and download links, and computation resources becomes mandatory. iii) Mobile users are battery-powered so that they are energy-constrained, or in other words, they have energy budgets. Without careful management, it would be possible that some of mobile users may use up their energy too fast to be available for any future participation. This may result in soaring maintenance cost as the remaining mobile users may ask for more reimbursements due to the reduction of competitions. By considering the above facts, the considered online incentive mechanism should be a nonlinear one. To address this nonlinearity, we first convert the original mechanism design problem into several one-shot design problems by temporally removing the energy constraint. Then, we propose a new mechanism design framework, i.e., IRSM, and based on that, design a new incentive mechanism for each one-shot problem. Finally, we reconsider energy constraints to design a new nonlinear online incentive mechanism by rationally combining the previously derived one-shot ones. Theoretical analyses show that our proposed nonlinear online incentive mechanism can guarantee individual rationality, truthfulness, a sound competitive ratio, and computational efficiency.

We further conduct comprehensive simulations to validate the effectiveness and superiority of our proposed mechanism.

In Chapter 5, to study the profits earned by service providers, a revenue maximization incentive mechanism design with budget constraints is studied in collaborative task offloading EC systems. The design aims to maximize the net revenue of the service provider and addresses more practical, but more complicated, scenarios of unknown *a prior* distribution information on mobile users' private information. To tackle this high computational complexity, which makes the traditional mechanism design methods infeasible, a new approach, called truthful deep mechanism, is proposed by leveraging a multi-task machine learning model, where inherently inter-connected collaborator selection and pricing policy determination are decided by designing two deep neural networks. The numerical results show that the proposed deep truthful mechanism can ensure a convergence to a stable state and can satisfy all required economical properties, including individual rationality, incentive compatibility, and budget balance.

6.2 Future work

Some future research directions on online incentive mechanism designs in wireless networks can be done in the following aspects.

- In practice, mobile users are battery-powered so that they have energy budgets. On one hand, such energy budgets could be reloaded by charging batteries with renewable energies (for example, solar and winding) or the wireless power transfer. While, on the other hand, energy budgets could also be reduced because mobile users may consume their own energies for playing games or watching videos. Thus, it would be more realistic and meaningful to design online incentive mechanisms by considering time-varying energy budgets. However, this problem would be extremely challenging due to the allocation issues. This is because in this scenario, mobile users have four actions, i.e., selected to execute tasks, recharging, discharging by playing games, no actions and waiting for future selections. The central controller should schedule this four different actions and their corresponding lasting time for all participants while still keeping enough potential mobile users for selections in order to achieve better competitive ratio. Furthermore, with the additional consideration of communication and computation resource allocation, the resource allocation issue would be more intractable. Intuitively, it could be possible to model the energy budget as a queue, and further apply queueing theory to analyze and solve this issue;

- Another potential work can be done on a nonlinear incentive mechanism design problem by considering the scenario where decisions must be made upon each single task arrival in wireless networks. However, most of the state-of-art online truthful mechanism designs for such a scenario are based on the primal-dual theory, which can only deal with linear objectives. Compared to linear online incentive mechanism designs, it is interesting, while challenging, to design online truthful mechanisms for applications whose objective and constraints are both nonlinear. The key and challenging part under this nonlinear one-by-one arrival scenario is how to devise an online allocation rule without future information. A possible solution thread is to utilize nonlinear online algorithm to get allocation solutions, and creatively design a pricing rule based on the allocation results;

- It is well-known that any mechanism design consists of two parts (i.e., resource allocation and pricing rule). For some complicated online incentive mechanism designs, such as nonlinear online incentive mechanisms, most existing methods could not be applied to this case. Thus, it would be difficult to design an allocation rule or pricing policy in a specific mathematical form. However, online incentive mechanisms for this issue can be designed by integrating recent advanced machine learning technologies, whereby we can apply online learning models, such as Q-learning, regularization-based learning, or online deep learning, to adaptively determine the allocation rule or pricing policy on-the-fly;

- Collusion in online mechanism designs should be emphasized. In reality, it is possible to collude bidders in a group in order to enlarge their benefits. Obviously, such collusion among bidders extremely complicates the online truthful mechanism design. A potential solution is to study indirect mechanism design in an online environment, such as online ascending incentive mechanism;

- Last but not least, privacy preservation is also important. Since mobile users are required to submit their private information to the central controller, some rival mobile users may have opportunities to intercept and eavesdrop that information or the untrustworthy central controller may tamper with mobile users' information. Such private information leakage or tampering will cause economic loss to those whose information is divulged or manipulated. Therefore, how to preserve privacy of mobile users is of great importance in reality and is an important issue in online truthful mechanism designs. Blockchain technology may be a possible solution to this issue, as it can find out whether submitted private information has been maliciously tempered with or not. On the other hand, we can avoid bids submission by devising distributed online truthful mechanisms so that no private information needs to be submitted to the central controller.

Appendix A

Appendix of Chapter 2

A.1 Proof of Lemma 2

We assume the final iteration in Algorithm 1 is the t-th iteration. Then we have

$$\begin{aligned}
\hat{w}_{i,j}^{(t)} &= \gamma(\mathcal{B}^{(t)})q_{i,j}^{(t)}(\mathcal{B}^{(t)}) = \hat{w}_{i,j}^{(t-1)} - \gamma(\mathcal{B}^{(t-1)})q_{i,j}^{(t-1)}(\mathcal{B}^{(t-1)}) \\
&= \hat{w}_{i,j}^{(t-2)} - \gamma(\mathcal{B}^{(t-2)})q_{i,j}^{(t-2)}(\mathcal{B}^{(t-2)}) - \gamma(\mathcal{B}^{(t-1)})q_{i,j}^{(t-1)}(\mathcal{B}^{(t-1)}) \\
&= w_{i,j} - \gamma(\mathcal{B}^{(1)})q_{i,j}^{(1)}(\mathcal{B}^{(1)}) - \cdots - \gamma(\mathcal{B}^{(t-1)})q_{i,j}^{(t-1)}(\mathcal{B}^{(t-1)}).
\end{aligned} \tag{121}$$

From (121), we get

$$\begin{aligned}
w_{i,j} &= \gamma(\mathcal{B}^{(1)})q_{i,j}^{(1)}(\mathcal{B}^{(1)}) + \cdots + \gamma(\mathcal{B}^{(t)})q_{i,j}^{(t)}(\mathcal{B}^{(t)}) \\
&= \sum_{\mathcal{B}\subseteq\mathcal{C}:(i,j)\notin\mathcal{B}} \gamma(\mathcal{B})q_{i,j}(\mathcal{B}).
\end{aligned} \tag{122}$$

Note that the equation (122) holds for any $(i,j) \in \mathcal{B}^{(t)}$. We consider the following two cases.

Case 1: $(i,j) \in \mathcal{B}^{(t)}$. From equation (122), we can always define a parameter α ($\alpha > 1$) such that

$$\frac{1}{\alpha} \sum_{\mathcal{B}\subseteq\mathcal{C}:(i,j)\notin\mathcal{B}} \gamma(\mathcal{B})q_{i,j}(\mathcal{B}) \leq w_{i,j}.$$

Case 2: $(i, j) \notin \mathcal{B}^{(t)}$. In this case, we first define

$$\alpha = \max_{i_1, i_2 \in \mathcal{M}, j_1, j_2 \in \mathcal{M}} \frac{q_{i_1,j_1}(\mathcal{B}) w_{i_2,j_2}}{q_{i_2,j_2}(\mathcal{B}) w_{i_1,j_1}}.$$

Then, we get

$$\frac{1}{\alpha} \frac{q_{i_1,j_1}(\mathcal{B})}{w_{i_1,j_1}} \leq \frac{q_{i_2,j_2}(\mathcal{B})}{w_{i_2,j_2}}, \; \forall (i_1, j_1) \notin \mathcal{B}^{(t)}, \forall (i_2, j_2) \in \mathcal{B}^{(t)}. \tag{123}$$

From inequality (123), we can derive that

$$\begin{aligned} \frac{1}{\alpha} \sum_{\mathcal{B} \subseteq \mathcal{C}:(i_1,j_1) \notin \mathcal{B}} \gamma(\mathcal{B}) \frac{q_{i_1,j_1}(\mathcal{B})}{w_{i_1,j_1}} &\leq \sum_{\mathcal{B} \subseteq \mathcal{C}:(i_2,j_2) \notin \mathcal{B}} \gamma(\mathcal{B}) \frac{q_{i_2,j_2}(\mathcal{B})}{w_{i_2,j_2}} = 1 \\ \Rightarrow \frac{1}{\alpha} \sum_{\mathcal{B} \subseteq \mathcal{C}:(i_1,j_1) \notin \mathcal{B}} \gamma(\mathcal{B}) q_{i_1,j_1}(\mathcal{B}) &\leq w_{i_1,j_1}. \end{aligned}$$

From the definition of α, we can observe that α is closely related to $q_{i,j}(\mathcal{B})$. Thus, after we check all the situations with the consideration of $q_{i,j}(\mathcal{B}) = \min\{q_{i,j}, \; Q(\mathcal{B})\}$ for any task ℓ, we can get

$$\frac{q_{i_1,j_1}(\mathcal{B})}{q_{i_2,j_2}(\mathcal{B})} = \max\{1, \; \frac{q_{i_1,j_1}}{q_{i_2,j_2}}\}.$$

Finally, we have

$$\alpha = \max_{i_1, i_2 \in \mathcal{M}, j_1, j_2 \in \mathcal{N}} \{\frac{w_{i_1,j_1}}{w_{i_2,j_2}}, \; \frac{w_{i_1,j_1} q_{i_2,j_2}}{w_{i_2,j_2} q_{i_1,j_1}}\}.$$

In summary, we can conclude that if α is chosen as the dual fitting factor, constraint C_9 can not be violated no matter whether the mobile users are selected or not, i.e., the Algorithm 1 can output the feasible solution to the dual problem (P6). This completes the proof of Lemma 2.

A.2 Proof of Theorem 2

The mobile users are selected by Algorithm 1 for each task, and the user is chosen while the constraint C_9 becomes tight with equality. Let p denote the solution to the primal problem (P2) by

Algorithm 1. We have

$$\begin{aligned}
p &= \sum_{i\in\mathcal{M}}\sum_{j\in\mathcal{N}} w_{i,j}x_{i,j} = \sum_{(i,j)\in\mathcal{B}^t} w_{i,j}x_{i,j} \\
&= \sum_{(i,j)\in\mathcal{B}^t}\ \sum_{\mathcal{B}\subseteq\mathcal{C}:(i,j)\notin\mathcal{B}} \gamma(\mathcal{A})q_{i,j}(\mathcal{A}) \\
&= \sum_{\mathcal{B}\subseteq\mathcal{C}} \gamma(\mathcal{B}) \sum_{(i,j)\in\mathcal{B}^t:(i,j)\notin\mathcal{B}} q_{i,j}(\mathcal{B}) \\
&\leq \sum_{\mathcal{B}\subseteq\mathcal{C}} \gamma(\mathcal{B})(\sum_{\mathcal{B}\subseteq\mathcal{B}^{(t-1)}} q_{i,j} - \sum_{(i,j)\in\mathcal{B}} q_{i,j} + q_{i^{'},j^{'}}(\mathcal{B})),
\end{aligned}$$

where $\mathcal{B}^{(t-1)}$ represents the collected bids at the $(t-1)$-th iteration, excluding the selected mobile user's bid $(i^{'},j^{'})$ at t-th iteration. Since $Q(\mathcal{B}^{(t-1)}) = Q - \sum_{(i,j)\in\mathcal{B}^{(t-1)}} q_{i,j} > 0$, we have $\sum_{(i,j)\in\mathcal{B}^{(t-1)}} q_{i,j} < Q$. Since $q_{i,j}(\mathcal{B}) \leq Q(\mathcal{B})$, $\forall i,\ j$, we have

$$\begin{aligned}
p &\leq \sum_{\mathcal{B}\subseteq\mathcal{C}} \gamma(\mathcal{B})(Q - \sum_{(i,j)\in\mathcal{B}} q_{i,j} + Q(\mathcal{B})) \\
&\leq \sum_{\mathcal{B}\subseteq\mathcal{C}} \gamma(\mathcal{B})(Q(\mathcal{B})) + Q(\mathcal{B}))) \\
&\leq 2\alpha \sum_{\mathcal{B}\subseteq\mathcal{C}} \frac{\gamma(\mathcal{B})}{\alpha} Q(\mathcal{B}) = 2\alpha d,
\end{aligned}$$

where d is the dual solution by Algorithm 1. Let p^* and p_f^* denote the optimal integer solution and the optimal fractional solution of problem (P2), respectively. Since $\frac{p}{d} \leq \beta$ holds, we have $\frac{p}{p^*} \leq \frac{p}{p_f^*} \leq \frac{p}{d} \leq \beta$ based on the linear duality theory [88]. This completes the proof for Theorem 2.

Appendix B

Appendix of Chapter 3

B.1 Proof of Lemma 8

Let Γ_{max}, Φ_{max}, and w_{max} be the maximum values of $\Gamma_{i,j}$, $\Phi_{i,j}$, and $w_{i,j}$, respectively, and r_{min} and ϕ_{min} be the minimum values of $r^k_{i,j}$ and $\phi_{i,j}$, respectively.

We first show that $\hat{p}^k_{j,t}$ can be bounded by the following expression:

$$\hat{p}^k_{j,t} \geq \frac{(1+\frac{1}{R^k_j})^{\sum\limits_{i\in\mathcal{U}'}\sum\limits_{\ell:t\in l^2_{i,j}(\ell)\in l^2_{i,j}} r^k_{i,j}x^\ell_{i,j}} - 1}{\Gamma_{max}}, \tag{124}$$

where $\mathcal{U}'$ denotes the set of all accepted requesters before requester i. We prove the above inequality through mathematical deduction. Define $\hat{p}^k_{j,t}(i)$ as the value of $\hat{p}^k_{j,t}$ before the arrival of requester i. At beginning, we have $x^\ell_{i,j} = 0\ \forall j, \ell$ and $\hat{p}^k_{j,t}(1) = 0$, so that inequality (124) holds. We then consider the following two cases:

- Case 1: Requester i is rejected by the BS. In this case, we have $x^\ell_{i,j} = 0$ and $p(i+1) = p(i)$. Obviously, the inequality (124) still holds, which does not affect the validation of $p^k_{j,t}(i+1)$.

- Case 2: Requester i is accepted by the BS. In this case, we have

$$
\begin{aligned}
&\hat{p}_{j,t}^k(i+1) = \hat{p}_{j,t}^k(i)(1+\frac{r_{i,j}^k}{R_j^k}) + \frac{r_{i,j}^k}{\Gamma_{i,j}R_j^k} \geq \hat{p}_{j,t}^k(i)(1+\frac{r_{i,j}^k}{R_j^k}) + \frac{r_{i,j}^k}{\Gamma_{max}R_j^k} \\
&\geq \frac{(1+\frac{1}{R_j^k})^{\sum\limits_{i\in\mathcal{U}'}\sum\limits_{\ell:t\in l_{i,j}^2(\ell)\in l_{i,j}^2} r_{i,j}^k x_{i,j}^\ell} - 1}{\Gamma_{max}}(1+\frac{r_{i,j}^k}{R_j^k}) + \frac{r_{i,j}^k}{\Gamma_{max}R_j^k} \\
&= \frac{(1+\frac{1}{R_j^k})^{\sum\limits_{i\in\mathcal{U}'}\sum\limits_{\ell:t\in l_{i,j}^2(\ell)\in l_{i,j}^2} r_{i,j}^k x_{i,j}^\ell}(1+\frac{r_{i,j}^k}{R_j^k})}{\Gamma_{max}} - \frac{1}{\Gamma_{max}} \\
&\approx \frac{(1+\frac{1}{R_j^k})^{\sum\limits_{i\in\mathcal{U}'}\sum\limits_{\ell:t\in l_{i,j}^2(\ell)\in l_{i,j}^2} r_{i,j}^k x_{i,j}^\ell}(1+\frac{1}{R_j^k})^{r_{i,j}^k}}{\Gamma_{max}} = \frac{(1+\frac{1}{R_j^k})^{\sum\limits_{i\in\mathcal{U}''}\sum\limits_{\ell:t\in l_{i,j}^2(\ell)\in l_{i,j}^2} r_{i,j}^k x_{i,j}^\ell}}{\Gamma_{max}},
\end{aligned}
\tag{125}
$$

  where $\mathcal{U}'' = \mathcal{U}' \cup i$ and the approximation holds because $R_j^k \gg 1$ and $r_{i,j}^k \ll R_j^k$, and $(1+a)^x \approx 1+ax$ when a and x are small enough.

Therefore, inequality (124) holds no matter whether requester i is accepted or not. However, $\hat{p}_{j,t}^k(+\infty) < \frac{w_{max}}{r_{min}}(1+1)+1 = 2\frac{w_{max}}{r_{min}}+1$ because of the conditions $w_{i,j} > \beta_{i,j}$ and $r_{i,j}^k \ll R_j^k$. By reconsidering the inequality (124), we have

$$
\frac{\sum\limits_{i\in\mathcal{U}'}\sum\limits_{\ell:t\in l_{i,j}^2(\ell)\in l_{i,j}^2} r_{i,j}^k x_{i,j}^\ell}{R_j^k} \leq \frac{\log(\Gamma_{max}(2\frac{w_{max}}{r_{min}}+1)+1)}{R_j^k\log(1+\frac{1}{R_j^k})} \approx \log(\Gamma_{max}(2\frac{w_{max}}{r_{min}}+1)+1), \tag{126}
$$

where the last approximation holds when $R_j^k \gg 1$. Inequality (126) indicates that the constraint C_{10} in problem (EQMSW) may be violated by at most $\log(\Gamma_{max}(2\frac{w_{max}}{r_{min}}+1)+1)$.

To verify that the solution meets constraint C_{13} in problem (EQMSW), we can follow the similar procedure to demonstrate that before the arrival of requester i, the value of $\hat{p}_t$ can be bounded as

$$
\hat{p}_t(i) \geq \frac{(1+\frac{1}{W})^{\sum\limits_{i\in\mathcal{U}'}\sum\limits_{j\in\mathcal{N}_i}\sum\limits_{\ell:t\in l_{i,j}^1(\ell)\in l_{i,j}^1} \phi_{i,j}x_{i,j}^\ell} - 1}{\Phi_{max}}. \tag{127}
$$

However, we have $\hat{p}_t(+\infty) < \frac{w_{max}}{\phi_{min}}(1+1)+1 = 2\frac{w_{max}}{\phi_{min}}+1$ because of the conditions $w_{i,j} > \beta_{i,j}$

and $\phi_{i,j} \ll W$. Combing those inequalities, we have

$$\frac{\sum\limits_{i\in\mathcal{U}'}\sum\limits_{j\in\mathcal{N}_i}\sum\limits_{\ell:t\in l^1_{i,j}(\ell)\in l^1_{i,j}} \phi_{i,j}x^{\ell}_{i,j}}{W} \leq \frac{\log(\Phi_{max}(2\frac{w_{max}}{\phi_{min}}+1)+1)}{W\log(1+\frac{1}{W})}$$
$$\approx \log(\Phi_{max}(2\frac{w_{max}}{\phi_{min}}+1)+1), \tag{128}$$

where the last approximation holds when $W \gg 1$. It indicates that the constraint C_6 in problem (EQMSW) may be violated by at most $\log(\Phi_{max}(2\frac{w_{max}}{\phi_{min}}+1)+1)$. This completes the proof.

B.2 Proof of Lemma 11

We first prove the truthfulness in the requesters' biding values. Note that the marginal prices $\hat{p}^k_{j,t}$ and $\hat{p}_t$ depend only on the past accepted requesters and are independent on the biding values of current requester i. Furthermore, the proposed online mechanism always assigns the requested resource to that requester only when the utility of that requester is maximized among all its bidding values and greater than zero given the current marginal prices. Therefore, our mechanism can be treated as a sequential posted price mechanism [120] or iterative auction [121], where the auctioneer posts the price and the bidders choose the best bidding values to maximize their utilities. In this way, the bidders cannot gain more utilities by misreporting their biding values.

Next, we demonstrate the truthfulness in arrival time t_i. If a requester reports the arrival time $t^{'}_i$ earlier than the actual value (i.e., $t^{'}_i < t_i$), this requester cannot increase its utility or even suffers from the failure to complete its task when $t^{'}_i < t_{i-1}$ or the transmission time is scheduled within the period of $[t^{'}_i, t_i]$. When the requester declares its arrival time later than t_i (i.e., $t^{'}_i > t_i$), the mechanism will find the optimal transmission and computation times after $t^{'}_i$ while in fact, such optimal times may happen in $[t_i, t^{'}_i]$, which results in an increased payment and a decreased utility. Thus, the requesters won't misreport their arrival time.

Third, it is obvious that the requesters won't intend to misreport their offloaded tasks (i.e., T_i) due to the fact that this can incur the failure completion of their tasks.

Finally, we verify the individual rationality. According to the acceptance condition (24), a requester can be accepted only if one of its maximum biddings can lead to a positive utility; otherwise, that requester is rejected and its utility is zero. Hence, our auction satisfies individual rationality. This completes the whole proof.

Appendix C

Appendix of Chapter 4

C.1 Proof of Lemma 12

It can be easily derived that constraints (74)-(77) and (79) are all convex. In the following, we mainly prove that the objective function is concave, and constraints (72), (73), and (78) are all convex.

- Concavity of the objective function.

  Since the function e^{kx} is convex with parameter k, $1 - e^{kx}$ is concave, indicating that the first term of objective function is concave. The second term is concave due to the fact that the summation of two convex functions and a linear function is still convex. Similarly, the last term is concave too. In summary, the objective function, which is the summation of three concave functions, is concave.

- Convexity of the constraint (78).

  Note that the first and the second terms in the constraint (78) have the same mathematical form of $f(x, y, z) = \frac{x^3}{yz}$, while the last term in the constraint (78) has the form of $g(x, y) = \frac{x^3}{y}$. Thus, we only need to prove $f(x, y, z)$ and $g(x, y)$ are convex. Calculate the second partial derivation of $f(x, y, z)$ with respect to x, y, z as

$$\Delta^2(x, y, z) = \begin{bmatrix} \frac{6x}{yz} & -3\frac{x^2y^2}{z} & -3\frac{x^2z^2}{y} \\ -3\frac{x^2y^2}{z} & 2\frac{x^3y^3}{z} & x^3(yz)^2 \\ -3\frac{x^2z^2}{y} & x^3(yz)^2 & 2\frac{x^3z^3}{y} \end{bmatrix} \tag{129}$$

Obviously, the first subdeterminant orders with respective to x, y, z are all less than zero. The second and the third subdeterminant orders of $\Delta^2(x, y, z)$ can be respectively calculated as

$$Det_2(\Delta^2(x, y, z)) = 3(xy)^4z^2 \geq 0,$$
$$Det_3(\Delta^2(x, y, z)) = 0.$$

Therefore, $f(x, y, z)$ is concave. Since $g(x, y)$ is a part of $f(x, y, z)$, it is easy to prove that its first subdeterminant order is less than zero, while its second subdeterminant order is no less than zero, which means $g(x, y)$ is concave. Since the summation of concave functions is still concave, the constraint (78) is concave.

- Convexity of constraints (72) and (73).

  After some simple manipulations, constraints (72) and (73) can be changed to $\alpha_{i,j}^D + \chi_{i,j}^3 - \ln z_{i,j}^D = 0$ and $\alpha_{i,j}^U + \chi_{i,j}^3 - \ln z_{i,j}^U = 0$, respectively. Thus, both (72) and (73) follow the same mathematical form of $f(x, y, z) = x^3 + y + \ln z^{-1}$. It is easy to prove that $f(x, y, z)$ is convex because a summation of two convex functions, i.e., x^3 and $\ln z^{-1}$, and a linear function is also convex. Therefore, both constraints (72) and (73) are convex.

Since the objective is concave and all constraints are convex, the optimization problem $[\mathcal{ERPO}1]$ is convex. This completes the proof.

Bibliography

[1] N. Poggi, "3 key internet of things trends to keep your eye on in 2017," Prey, Inc, Discussion Poster, Jun. 2017. [Online]. Available: https://preyproject.com/blog/en/3-key-internet-of-things-trends-to-keep-your-eye-on-in-2017/

[2] T. Luo, S. S. Kanhere, J. Huang, S. K. Das, and F. Wu, "Sustainable incentives for mobile crowdsensing: Auctions, lotteries, and trust and reputation systems," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 68–74, 2017.

[3] W. Vickrey, "Counterspeculation, auctions, and competitive sealed tenders," *The Journal of Finance*, vol. 16, no. 1, pp. 8–37, 1961. [Online]. Available: http://www.jstor.org/stable/2977633

[4] D. Ferguson, Y. Yemini, and C. Nikolaou, "Microeconomic algorithms for load balancing in distributed computer systems," in *[1988] Proceedings. The 8th International Conference on Distributed*, 1988, pp. 491–499.

[5] N. Amitay, "Resource auction multiple access (rama): efficient method for fast resource assignment in decentralised wireless pcs," *Electronics Letters*, vol. 28, pp. 799–801(2), April 1992. [Online]. Available: https://digital-library.theiet.org/content/journals/10.1049/el_19920504

[6] A. R. Karlin, M. S. Manasse, L. Rudolph, and D. D. Sleator, "Competitive snoopy caching," in *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*, 1986, pp. 244–254.

[7] R. Lavi and C. Swamy, "Truthful and near-optimal mechanism design via linear programming," *J. ACM*, vol. 58, no. 6, Dec. 2011. [Online]. Available: https://doi.org/10.1145/2049697.2049699

[8] Y. Bartal, R. Gonen, and N. Nisan, "Incentive compatible multi unit combinatorial auctions," in *Proceedings of the 9th Conference on Theoretical Aspects of Rationality and Knowledge*, ser. TARK '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 7287. [Online]. Available: https://doi.org/10.1145/846241.846250

[9] E. J. Friedman and D. C. Parkes, "Pricing wifi at starbucks: Issues in online mechanism design," in *Proceedings of the 4th ACM Conference on Electronic Commerce*, ser. EC '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 240241. [Online]. Available: https://doi.org/10.1145/779928.779978

[10] D. C. Parkes and S. Singh, "An mdp-based approach to online mechanism design," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds., vol. 16. MIT Press, 2004. [Online]. Available: https://proceedings.neurips.cc/paper/2003/file/d16509f6eaca1022bd8f28d6bc582cae-Paper.pdf

[11] J. Bredin and D. C. Parkes, "Models for truthful online double auctions," *CoRR*, vol. abs/1207.1360, 2012. [Online]. Available: http://arxiv.org/abs/1207.1360

[12] A. Blum, T. Sandholm, and M. Zinkevich, "Online algorithms for market clearing," *J. ACM*, vol. 53, no. 5, p. 845879, Sep. 2006. [Online]. Available: https://doi.org/10.1145/1183907.1183913

[13] J. Gallien, "Dynamic mechanism design for online commerce," *Oper. Res.*, vol. 54, no. 2, p. 291310, Mar. 2006. [Online]. Available: https://doi.org/10.1287/opre.1050.0239

[14] M. Pai and R. Vohra, "Optimal Dynamic Auctions," Northwestern University, Center for Mathematical Studies in Economics and Management Science, Discussion Papers 1461, Mar. 2008. [Online]. Available: https://ideas.repec.org/p/nwu/cmsems/1461.html

[15] N. Buchbinder and R. Gonen, *Incentive Compatible Mulit-Unit Combinatorial Auctions: A Primal Dual Approach*. Berlin, Heidelberg: Springer-Verlag, May 2015, vol. 72, no. 1, p. 167190. [Online]. Available: https://doi.org/10.1007/s00453-013-9854-4

[16] H. Li, C. Wu, and Z. Li, "Socially-optimal online spectrum auctions for secondary wireless communication," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 2047–2055.

[17] Y. Zhang, M. Pan, L. Song, Z. Dawy, and Z. Han, "A survey of contract theory-based incentive mechanism design in wireless networks," *IEEE Wireless Communications*, vol. 24, no. 3, pp. 80–85, 2017.

[18] G. Chatzimilioudis, A. Konstantinidis, C. Laoudias, and D. Zeinalipour-Yazti, "Crowdsourcing with smartphones," *IEEE Internet Computing*, vol. 16, no. 5, pp. 36–44, 2012.

[19] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda, "Peir, the personal environmental impact report, as a platform for participatory sensing systems research," in *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 5568. [Online]. Available: https://doi.org/10.1145/1555816.1555823

[20] Y. Zheng, L. Li, and L. Zhang, "Poster abstract: Pimi air community: Getting fresher indoor air by sharing data and know-hows," in *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, 2014, pp. 283–284.

[21] B. Pan, Y. Zheng, D. Wilkie, and C. Shahabi, "Crowd sensing of traffic anomalies based on human mobility and social media," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ser. SIGSPATIAL'13. New York, NY, USA: Association for Computing Machinery, 2013, p. 344353. [Online]. Available: https://doi.org/10.1145/2525314.2525343

[22] X. Zhu, J. An, M. Yang, L. Xiang, Q. Yang, and X. Gui, "A fair incentive mechanism for crowdsourcing in crowd sensing," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1364–1372, 2016.

[23] C. Tang, X. Li, M. Cao, Z. Zhang, and X. Yu, "Incentive mechanism for macrotasking crowdsourcing: A zero-determinant strategy approach," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8589–8601, 2019.

[24] J. Li, Y. Zhu, Y. Hua, and J. Yu, "Crowdsourcing sensing to smartphones: A randomized auction approach," *IEEE Transactions on Mobile Computing*, vol. 16, no. 10, pp. 2764–2777, 2017.

[25] Z. Feng, Y. Zhu, Q. Zhang, L. M. Ni, and A. V. Vasilakos, "Trac: Truthful auction for location-aware collaborative sensing in mobile crowdsourcing," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, 2014, pp. 1231–1239.

[26] H. Jin, L. Su, D. Chen, K. Nahrstedt, and J. Xu, "Quality of information aware incentive mechanisms for mobile crowd sensing systems," in *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 167176. [Online]. Available: https://doi.org/10.1145/2746285.2746310

[27] J. Xu, J. Xiang, and D. Yang, "Incentive mechanisms for time window dependent tasks in mobile crowdsensing," *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 6353–6364, 2015.

[28] D. Peng, F. Wu, and G. Chen, "Data quality guided incentive mechanism design for crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 17, no. 2, pp. 307–319, 2018.

[29] X. Zhang, Z. Yang, Z. Zhou, H. Cai, L. Chen, and X. Li, "Free market of crowdsourcing: Incentive mechanism design for mobile sensing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 12, pp. 3190–3200, 2014.

[30] A. Subramanian, G. S. Kanth, S. Moharir, and R. Vaze, "Online incentive mechanism design for smartphone crowd-sourcing," in *2015 13th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 2015, pp. 403–410.

[31] D. Zhao, X. Li, and H. Ma, "Budget-feasible online incentive mechanisms for crowdsourcing tasks truthfully," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 647–661, 2016.

[32] L. Pu, X. Chen, J. Xu, and X. Fu, "Crowd foraging: A qos-oriented self-organized mobile crowdsourcing framework over opportunistic networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 4, pp. 848–862, 2017.

[33] R. Zhou, Z. Li, and C. Wu, "A truthful online mechanism for location-aware tasks in mobile crowd sensing," *IEEE Transactions on Mobile Computing*, vol. 17, no. 8, pp. 1737–1749, 2018.

[34] X. Tian, W. Zhang, Y. Yang, X. Wu, Y. Peng, and X. Wang, "Toward a quality-aware online pricing mechanism for crowdsensed wireless fingerprints," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 5953–5964, 2018.

[35] W. Gong, B. Zhang, and C. Li, "Location-based online task assignment and path planning for mobile crowdsensing," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1772–1783, 2019.

[36] L. Chen, J. Xu, and S. Zhou, "Computation peer offloading in mobile edge computing with energy budgets," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, 2017, pp. 1–6.

[37] R. B. Myerson, "Optimal auction design," *Mathematics of Operations Research*, vol. 6, no. 1, pp. 58–73, 1981. [Online]. Available: http://www.jstor.org/stable/3689266

[38] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic Game Theory*. USA: Cambridge University Press, 2007.

[39] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5g be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.

[40] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, *Mobile Edge Computing A key technology towards 5G*. France, Sophia Antipolis CEDEX: European Telecommunications Standards Institute, 2015.

[41] X. Chen, L. Pu, L. Gao, W. Wu, and D. Wu, "Exploiting massive d2d collaboration for energy-efficient mobile edge computing," *IEEE Wireless Communications*, vol. 24, no. 4, pp. 64–71, 2017.

[42] Y. He, J. Ren, G. Yu, and Y. Cai, "D2d communications meet mobile edge computing for enhanced computation capacity in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 3, pp. 1750–1763, 2019.

[43] X. Cao, F. Wang, J. Xu, R. Zhang, and S. Cui, "Joint computation and communication cooperation for energy-efficient mobile edge computing," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4188–4200, 2019.

[44] Y. Cui, J. Song, K. Ren, M. Li, Z. Li, Q. Ren, and Y. Zhang, "Software defined cooperative offloading for mobile cloudlets," *IEEE/ACM Transactions on Networking*, vol. 25, no. 3, pp. 1746–1760, 2017.

[45] N. T. Ti and L. B. Le, "Computation offloading leveraging computing resources from edge cloud and mobile peers," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6.

[46] A. Kiani and N. Ansari, "Toward hierarchical mobile edge computing: An auction-based profit maximization approach," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 2082–2091, 2017.

[47] H. Zhang, F. Guo, H. Ji, and C. Zhu, "Combinational auction-based service provider selection in mobile edge computing networks," *IEEE Access*, vol. 5, pp. 13 455–13 464, 2017.

[48] Y. Jiao, P. Wang, D. Niyato, and K. Suankaewmanee, "Auction mechanisms in cloud/fog computing resource allocation for public blockchain networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 9, pp. 1975–1989, 2019.

[49] A.-L. Jin, W. Song, and W. Zhuang, "Auction-based resource allocation for sharing cloudlets in mobile cloud computing," *IEEE Transactions on Emerging Topics in Computing*, vol. 6, no. 1, pp. 45–57, 2018.

[50] 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 of Things Journal*, vol. 6, no. 3, pp. 4214–4227, 2019.

[51] Y. Zhang, L. Song, W. Saad, Z. Dawy, and Z. Han, "Contract-based incentive mechanisms for device-to-device communications in cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 10, pp. 2144–2155, 2015.

[52] T. Wang, Y. Sun, L. Song, and Z. Han, "Social data offloading in d2d-enhanced cellular networks by network formation games," *IEEE Transactions on Wireless Communications*, vol. 14, no. 12, pp. 7004–7015, 2015.

[53] L. Yang, H. Zhang, M. Li, J. Guo, and H. Ji, "Mobile edge computing empowered energy efficient task offloading in 5g," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 7, pp. 6398–6409, 2018.

[54] Q. Wang, S. Guo, J. Liu, C. Pan, and L. Yang, "Profit maximization incentive mechanism for resource providers in mobile edge computing," *IEEE Transactions on Services Computing*, pp. 1–1, 2019.

[55] G. Li and J. Cai, "An online incentive mechanism for collaborative task offloading in mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 624–636, 2020.

[56] J. He, D. Zhang, Y. Zhou, and Y. Zhang, "A truthful online mechanism for collaborative computation offloading in mobile edge computing," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4832–4841, 2020.

[57] D. Zhang, L. Tan, J. Ren, M. K. Awad, S. Zhang, Y. Zhang, and P.-J. Wan, "Near-optimal and truthful online auction for computation offloading in green edge-computing systems," *IEEE Transactions on Mobile Computing*, vol. 19, no. 4, pp. 880–893, 2020.

[58] T. Sandholm and A. Likhodedov, "Automated design of revenue-maximizing combinatorial auctions," *Oper. Res.*, vol. 63, no. 5, p. 10001025, Oct. 2015.

[59] A. Chi-Chih Yao, "On Solutions for the Maximum Revenue Multi-item Auction under Dominant-Strategy and Bayesian Implementations," *arXiv e-prints*, p. arXiv:1607.03685, Jul. 2016.

[60] M. Al-Ayyoub and H. Gupta, "Truthful spectrum auctions with approximate social-welfare or revenue," *IEEE/ACM Transactions on Networking*, vol. 22, no. 6, pp. 1873–1885, 2014.

[61] C. Avery and T. Hendershott, "Bundling and optimal auctions of multiple products," *The Review of Economic Studies*, vol. 67, no. 3, pp. 483–497, 07 2000. [Online]. Available: https://doi.org/10.1111/1467-937X.00140

[62] Z. Zheng, F. Wu, X. Gao, H. Zhu, S. Tang, and G. Chen, "A budget feasible incentive mechanism for weighted coverage maximization in mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 16, no. 9, pp. 2392–2407, 2017.

[63] C.-X. Wang, M. D. Renzo, S. Stanczak, S. Wang, and E. G. Larsson, "Artificial intelligence enabled wireless networking for 5g and beyond: Recent advances and future challenges," *IEEE Wireless Communications*, vol. 27, no. 1, pp. 16–23, 2020.

[64] Z. M. Fadlullah, F. Tang, B. Mao, N. Kato, O. Akashi, T. Inoue, and K. Mizutani, "State-of-the-art deep learning: Evolving machine intelligence toward tomorrows intelligent network traffic control systems," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2432–2455, 2017.

[65] N. Kato, Z. M. Fadlullah, B. Mao, F. Tang, O. Akashi, T. Inoue, and K. Mizutani, "The deep learning vision for heterogeneous network traffic control: Proposal, challenges, and future perspective," *IEEE Wireless Communications*, vol. 24, no. 3, pp. 146–153, 2017.

[66] T. Q. Dinh, Q. D. La, T. Q. S. Quek, and H. Shin, "Learning for computation offloading in mobile edge computing," *IEEE Transactions on Communications*, vol. 66, no. 12, pp. 6353–6367, 2018.

[67] C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, M. Yunsheng, S. Chen, and P. Hou, "A new deep learning-based food recognition system for dietary assessment on an edge computing service infrastructure," *IEEE Transactions on Services Computing*, vol. 11, no. 2, pp. 249–261, 2018.

[68] Y. Liu, H. Yu, S. Xie, and Y. Zhang, "Deep reinforcement learning for offloading and resource allocation in vehicle edge computing and networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11 158–11 168, 2019.

[69] L. Huang, S. Bi, and Y.-J. A. Zhang, "Deep reinforcement learning for online computation offloading in wireless powered mobile-edge computing networks," *IEEE Transactions on Mobile Computing*, vol. 19, no. 11, pp. 2581–2593, 2020.

[70] M. Hu, L. Zhuang, D. Wu, Y. Zhou, X. Chen, and L. Xiao, "Learning driven computation offloading for asymmetrically informed edge computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 8, pp. 1802–1815, 2019.

[71] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. S. Quek, "Offloading in mobile edge computing: Task allocation and computational frequency scaling," *IEEE Transactions on Communications*, vol. 65, no. 8, pp. 3571–3584, 2017.

[72] Y. Yu, J. Zhang, and K. B. Letaief, "Joint subcarrier and cpu time allocation for mobile edge computing," in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016, pp. 1–6.

[73] Y. Mao, J. Zhang, and K. B. Letaief, “Dynamic computation offloading for mobile-edge computing with energy harvesting devices,” *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3590–3605, 2016.

[74] Z. Zhou, Q. Wu, and X. Chen, “Online orchestration of cross-edge service function chaining for cost-efficient edge computing,” *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 8, pp. 1866–1880, 2019.

[75] Y. Cai, C. Daskalakis, and S. M. Weinberg, “An algorithmic characterization of multi-dimensional mechanisms,” in *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*, ser. STOC ’12. New York, NY, USA: Association for Computing Machinery, 2012, p. 459478. [Online]. Available: https://doi.org/10.1145/2213977.2214021

[76] S. Bhattacharya, G. Goel, S. Gollapudi, and K. Munagala, “Budget constrained auctions with heterogeneous items,” in *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, ser. STOC ’10. New York, NY, USA: Association for Computing Machinery, 2010, p. 379388. [Online]. Available: https://doi.org/10.1145/1806689.1806743

[77] Y. Cai, C. Daskalakis, and S. Weinberg, “Optimal multi-dimensional mechanism design: Reducing revenue to welfare maximization,” *Proceedings - Annual IEEE Symposium on Foundations of Computer Science, FOCS*, pp. 130–139, 2012, 53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012 ; Conference date: 20-10-2012 Through 23-10-2012.

[78] C. Daskalakis, N. R. Devanur, and S. M. Weinberg, “Revenue maximization and ex-post budget constraints,” *ACM Trans. Econ. Comput.*, vol. 6, no. 34, Oct. 2018. [Online]. Available: https://doi.org/10.1145/3274647

[79] Y. Cai, C. Daskalakis, and S. Weinberg, “Understanding incentives: Mechanism design becomes algorithm design,” in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*. Los Alamitos, CA, USA: IEEE Computer Society, oct 2013, pp. 618–627. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/FOCS.2013.72

[80] A. C. Yao, “On solutions for the maximum revenue multi-item auction under dominant-strategy and bayesian implementations,” *CoRR*, vol. abs/1607.03685, 2016. [Online]. Available: http://arxiv.org/abs/1607.03685

[81] N. C. Luong, Z. Xiong, P. Wang, and D. Niyato, “Optimal auction for edge computing resource management in mobile blockchain networks: A deep learning approach,” in *2018 IEEE International Conference on Communications (ICC)*, 2018, pp. 1–6.

[82] S. Yang, F. Wu, S. Tang, X. Gao, B. Yang, and G. Chen, “On designing data quality-aware truth estimation and surplus sharing method for mobile crowdsensing,” *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 4, pp. 832–847, 2017.

[83] H. Li, B. Zhao, and A. Fuxman, “The wisdom of minority: Discovering and targeting the right group of workers for crowdsourcing,” in *Proceedings of the 23rd International Conference on World Wide Web*, ser. WWW ’14. New York, NY, USA: Association for Computing Machinery, 2014, p. 165176. [Online]. Available: https://doi.org/10.1145/2566486.2568033

[84] T. Carnes and D. Shmoys, “Primal-dual schema for capacitated covering problems,” in *Integer Programming and Combinatorial Optimization*, A. Lodi, A. Panconesi, and G. Rinaldi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 288–302.

[85] R. D. Carr, L. K. Fleischer, V. J. Leung, and C. A. Phillips, “Strengthening integrality gaps for capacitated network design and covering problems,” in *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA ’00. USA: Society for Industrial and Applied Mathematics, 2000, p. 106115.

[86] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[87] L. Khachiyan, “Polynomial algorithms in linear programming,” *USSR Computational Mathematics and Mathematical Physics*, vol. 20, no. 1, pp. 53–72, 1980. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0041555380900610

[88] V. V. Vazirani, *Approximation Algorithms*. Berlin, Heidelberg: Springer-Verlag, 2001.

[89] S. Yadav and A. Malik, *Operations Research*, ser. Oxford higher education. Oxford University Press, 2014. [Online]. Available: https://books.google.ca/books?id=cZAKjwEACAAJ

[90] S. Gu, Z. Li, C. Wu, and H. Zhang, "Virtualized resource sharing in cloud radio access networks through truthful mechanisms," *IEEE Transactions on Communications*, vol. 65, no. 3, pp. 1105–1118, 2017.

[91] A. Archer and E. Tardos, "Truthful mechanisms for one-parameter agents," in *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, 2001, pp. 482–491.

[92] C. Wang, C. Liang, F. R. Yu, Q. Chen, and L. Tang, "Computation offloading and resource allocation in wireless cellular networks with mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 16, no. 8, pp. 4924–4938, 2017.

[93] C. You, K. Huang, H. Chae, and B. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1397–1411, 2017.

[94] S. He, D. Shin, J. Zhang, J. Chen, and P. Lin, "An exchange market approach to mobile crowdsensing: Pricing, task allocation, and walrasian equilibrium," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 4, pp. 921–934, 2017.

[95] B. Fan, H. Tian, L. Jiang, and A. V. Vasilakos, "A social-aware virtual mac protocol for energy-efficient d2d communications underlying heterogeneous cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 9, pp. 8372–8385, 2018.

[96] K. Doppler, M. Rinne, C. Wijting, C. B. Ribeiro, and K. Hugl, "Device-to-device communication as an underlay to lte-advanced networks," *IEEE Communications Magazine*, vol. 47, no. 12, pp. 42–49, 2009.

[97] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795–2808, 2016.

[98] D. Huang, P. Wang, and D. Niyato, “A dynamic offloading algorithm for mobile computing,” *IEEE Transactions on Wireless Communications*, vol. 11, no. 6, pp. 1991–1995, 2012.

[99] Y. Wen, W. Zhang, and H. Luo, “Energy-optimal mobile application execution: Taming resource-poor mobile devices with cloud clones,” in *2012 Proceedings IEEE INFOCOM*, 2012, pp. 2716–2720.

[100] J. Borghoff, L. R. Knudsen, and M. Stolpe, “Bivium as a mixed-integer linear programming problem,” in *Cryptography and Coding*, M. G. Parker, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 133–152.

[101] M.-H. Chen, B. Liang, and M. Dong, “Joint offloading and resource allocation for computation and communication in mobile cloud with computing access point,” in *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017, pp. 1–9.

[102] F. Guo, L. Ma, H. Zhang, H. Ji, and X. Li, “Joint load management and resource allocation in the energy harvesting powered small cell networks with mobile edge computing,” in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2018, pp. 299–304.

[103] D. Irwin, L. Grit, and J. Chase, “Balancing risk and reward in a market-based task service,” in *Proceedings. 13th IEEE International Symposium on High performance Distributed Computing, 2004.*, 2004, pp. 160–169.

[104] R. Lavi and N. Nisan, “Competitive analysis of incentive compatible on-line auctions,” in *Proceedings of the 2nd ACM Conference on Electronic Commerce*, ser. EC ’00. New York, NY, USA: Association for Computing Machinery, 2000, p. 233241. [Online]. Available: https://doi.org/10.1145/352871.352897

[105] C. Yi, S. Huang, and J. Cai, “An incentive mechanism integrating joint power, channel and link management for social-aware d2d content sharing and proactive caching,” *IEEE Transactions on Mobile Computing*, vol. 17, no. 4, pp. 789–802, 2018.

[106] P. Wang, Z. Zheng, B. Di, and L. Song, "Hetmec: Latency-optimal task assignment and resource allocation for heterogeneous multi-layer mobile edge computing," *IEEE Transactions on Wireless Communications*, vol. 18, no. 10, pp. 4942–4956, 2019.

[107] J. Zhang, X. Huang, and R. Yu, "Optimal task assignment with delay constraint for parked vehicle assisted edge computing: A stackelberg game approach," *IEEE Communications Letters*, vol. 24, no. 3, pp. 598–602, 2020.

[108] L. Pu, X. Chen, J. Xu, and X. Fu, "D2d fogging: An energy-efficient and incentive-aware task offloading framework via network-assisted d2d collaboration," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3887–3901, 2016.

[109] R. Rajaraman, *Randomized Rounding*, M.-Y. Kao, Ed. Boston, MA: Springer US, 2008. [Online]. Available: https://doi.org/10.1007/978-0-387-30162-4_327

[110] S. Dughmi, T. Roughgarden, and Q. Yan, "Optimal mechanisms for combinatorial auctions and combinatorial public projects via convex rounding," *J. ACM*, vol. 63, no. 4, Sep. 2016. [Online]. Available: https://doi.org/10.1145/2908735

[111] N. Nisan and A. Ronen, "Computationally Feasible VCG Mechanisms," *arXiv e-prints*, p. arXiv:1110.0025, sep 2011.

[112] S. Dobzinski and S. Dughmi, "On the power of randomization in algorithmic mechanism design," in *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, ser. FOCS '09. USA: IEEE Computer Society, 2009, p. 505514. [Online]. Available: https://doi.org/10.1109/FOCS.2009.42

[113] G. Li and J. Cai, "An online mechanism for crowdsensing with uncertain task arriving," in *2018 IEEE International Conference on Communications (ICC)*, 2018, pp. 1–6.

[114] H. Yu, H. D. Tuan, T. Q. Duong, H. V. Poor, and Y. Fang, "Optimization for signal transmission and reception in a macrocell of heterogeneous uplinks and downlinks," *IEEE Transactions on Communications*, vol. 68, no. 11, pp. 7054–7067, 2020.

[115] A. Gopinathan, N. Carlsson, Z. Li, and C. Wu, "Revenue-maximizing and truthful online auctions for dynamic spectrum access," in *2016 12th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, 2016, pp. 1–8.

[116] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[117] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods (Optimization and Neural Computation Series)*, 1st ed. Athena Scientific, 1996. [Online]. Available: http://www.amazon.com/Constrained-Optimization-Lagrange-Multiplier-Computation/dp/1886529043%3FSubscriptionId%3D192BW6DQ43CK9FN0ZGG2%26tag%3Dws%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3D1886529043

[118] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Learning Representations by Back-Propagating Errors*. Cambridge, MA, USA: MIT Press, 1988, p. 696699.

[119] E. Li, L. Zeng, Z. Zhou, and X. Chen, "Edge ai: On-demand accelerating deep neural network inference via edge computing," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 447–457, 2020.

[120] S. Chawla, J. D. Hartline, D. L. Malec, and B. Sivan, "Multi-parameter mechanism design and sequential posted pricing," in *Proceedings of the Forty-Second ACM Symposium on Theory of Computing*, ser. STOC '10. New York, NY, USA: Association for Computing Machinery, 2010, p. 311320. [Online]. Available: https://doi.org/10.1145/1806689.1806733

[121] G. Iosifidis, L. Gao, J. Huang, and L. Tassiulas, "A double-auction mechanism for mobile data-offloading markets," *IEEE/ACM Transactions on Networking*, vol. 23, no. 5, pp. 1634–1647, 2015.

List of Publications

[1] (Book Chapter), G. Li and J. Cai, Online Incentive Mechanism Design in Edge Computing Springer, 2021. (In Press)

[2] G. Li and J. Cai, ”An Online Incentive Mechanism for Collaborative Task Offloading in Mobile Edge Computing,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 624-636, Jan. 2020.

[3] G. Li and J. Cai, ”An Online Incentive Mechanism for Crowdsensing with Random Task Arrivals, *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2982-2995, April 2020.

[4] G. Li and J. Cai, ” Online Incentive Mechanism Design in Wireless Communications,” *IEEE Wireless Communication*, vol. 28, no. 4, pp. 159-165, Aug. 2021.

[5] G. Li, J. Cai, and S. Ni, “A Truthful Deep Mechanism Design with Budget Constraints for Revenue-Maximization in Edge Computing”, Accepted in *IEEE Trans. Veh. Technol.*

[6] G. Li, H. Chen, and J. Cai, ”Joint User Association and Power Allocation for Hybrid Half-Duplex/Full-Duplex Relaying in Cellular Networks,” *IEEE Systems Journal*, vol. 13, no. 2, pp. 1145-1156, June 2019.

[7] G. Li, J. Cai, and J. M, ”Online Incentive Mechanism Design for Collaborative Offloading in Mobile Edge Computing,” IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 2019, pp. 1-6.

[8] G. Li and J. Cai, ”An Online Mechanism for Crowdsensing with Uncertain Task Arriving,” IEEE International Conference on Communications (ICC), Kansas City, MO, 2018, pp. 1-6.

[9] G. Li, J. Cai, X. Chen, and Z. Su, "Nonlinear Online Incentive Mechanism Design in Edge Computing Systems with Energy Budget", Submitted to *IEEE Trans. Mobile Comput.* (Major Revision)