Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing

Xu Chen, *Member, IEEE*, Lei Jiao, *Member, IEEE*, Wenzhong Li, *Member, IEEE, ACM*, and Xiaoming Fu, *Senior Member, IEEE*

***Abstract*—Mobile-edge cloud computing is a new paradigm to provide cloud computing capabilities at the edge of pervasive radio access networks in close proximity to mobile users. In this paper, we first study the multi-user computation offloading problem for mobile-edge cloud computing in a multi-channel wireless interference environment. We show that it is NP-hard to compute a centralized optimal solution, and hence adopt a game theoretic approach for achieving efficient computation offloading in a distributed manner. We formulate the distributed computation offloading decision making problem among mobile device users as a multi-user computation offloading game. We analyze the structural property of the game and show that the game admits a Nash equilibrium and possesses the finite improvement property. We then design a distributed computation offloading algorithm that can achieve a Nash equilibrium, derive the upper bound of the convergence time, and quantify its efficiency ratio over the centralized optimal solutions in terms of two important performance metrics. We further extend our study to the scenario of multi-user computation offloading in the multi-channel wireless contention environment. Numerical results corroborate that the proposed algorithm can achieve superior computation offloading performance and scale well as the user size increases.**

***Index Terms*—Computation offloading, game theory, mobile-edge cloud computing, Nash equilibrium.**

I. Introduction

As SMARTPHONES are gaining enormous popularity, more and more new mobile applications such as face recognition, natural language processing, interactive gaming, and augmented reality are emerging and attract great attention [1]–[3]. This kind of mobile applications are typically resource-hungry, demanding intensive computation and high energy consumption. Due to the physical size constraint, however, mobile devices are in general resource-constrained, having limited computation resources and battery life. The tension between resource-hungry applications and resource-constrained mobile devices hence poses a significant challenge for the future mobile platform development [4].

Mobile cloud computing is envisioned as a promising approach to address such a challenge. By offloading the computation via wireless access to the resource-rich cloud infrastructure, mobile cloud computing can augment the capabilities of mobile devices for resource-hungry applications. One possible approach is to offload the computation to the remote public clouds such as Amazon EC2 and Windows Azure. However, an evident weakness of public cloud based mobile cloud computing is that mobile users may experience long latency for data exchange with the public cloud through the wide area network. Long latency would hurt the interactive response, since humans are acutely sensitive to delay and jitter. Moreover, it is very difficult to reduce the latency in the wide area network. To overcome this limitation, the cloudlet based mobile cloud computing was proposed as a promising solution [5]. Rather than relying on a remote cloud, the cloudlet based mobile cloud computing leverages the physical proximity to reduce delay by offloading the computation to the nearby computing sever/cluster via one-hop WiFi wireless access. However, there are two major disadvantages for the cloudlet based mobile cloud computing: 1) due to limited coverage of WiFi networks (typically available for indoor environments), cloudlet based mobile cloud computing can not guarantee ubiquitous service provision everywhere; 2) due to space constraint, cloudlet based mobile cloud computing usually utilizes a computing sever/cluster with small/medium computation resources, which may not satisfy QoS of a large number of users.

To address these challenges and complement cloudlet based mobile cloud computing, a novel mobile cloud computing paradigm, called mobile-edge cloud computing, has been proposed [6]–[9]. As illustrated in Fig. 1, mobile-edge cloud computing can provide cloud-computing capabilities at the edge of pervasive radio access networks in close proximity to mobile users. In this case, the need for fast interactive response can be met by fast and low-latency connection (e.g., via fiber transmission) to large-scale resource-rich cloud computing infrastructures (called telecom cloud) deployed by telecom operators (e.g., AT&T and T-Mobile) within the network edge and backhaul/core networks. By endowing ubiquitous radio access networks (e.g., 3G/4G macro-cell and small-cell base-stations) with powerful computing capabilities, mobile-edge cloud computing is envisioned to provide pervasive and agile computation

Manuscript received March 17, 2015; revised September 05, 2015; accepted September 29, 2015; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor U. Ayesta. Date of publication October 26, 2015; date of current version October 13, 2016. This work was supported in part by the EU FP7 IRSES MobileCloud Project under Grant No. 612212, the National Natural Science Foundation of China under Grants No. 61373128 and No. 61321491, the Sino-German Institutes of Social Computing, the Simulation Science Center sponsored by State Lower Saxony and Volkswagen Foundation, and the Alexander von Humboldt Foundation. *(Corresponding author: Wenzhong Li.)*

X. Chen and X. Fu are with the Institute of Computer Science, University of Göttingen, 37077 Göttingen, Germany (e-mail: xu.chen@cs.uni-goettingen.de; fu@cs.uni-goettingen.de).

L. Jiao was with the University of Göttingen, 37077 Göttingen, Germany. He is now with Bell Labs, Alcatel-Lucent, Dublin, Ireland (e-mail: lei.jiao@bell-labs.com).

W. Li is with the State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China (e-mail: lwz@nju.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNET.2015.2487344

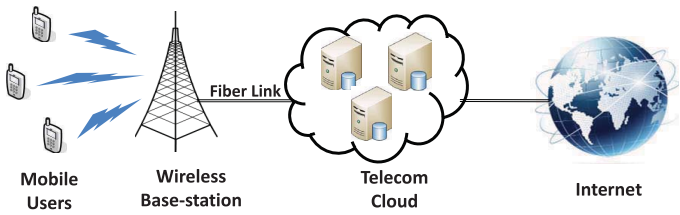


Fig. 1. An illustration of mobile-edge cloud computing.

augmenting services for mobile device users at anytime and anywhere [6]–[9].

In this paper, we study the issue of designing efficient computation offloading mechanism for mobile-edge cloud computing. One critical factor of affecting the computation offloading performance is the wireless access efficiency [10]. Given the fact that base-stations in most wireless networks are operating in multi-channel setting, a key challenge is how to achieve efficient wireless access coordination among multiple mobile device users for computation offloading. If too many mobile device users choose the same wireless channel to offload the computation to the cloud simultaneously, they may cause severe interference to each other, which would reduce the data rates for computation offloading. This hence can lead to low energy efficiency and long data transmission time. In this case, it would not be beneficial for the mobile device users to offload computation to the cloud. To achieve efficient computation offloading for mobile-edge cloud computing, we hence need to carefully tackle two key challenges: 1) how should a mobile user choose between the local computing (on its own device) and the cloud computing (via computation offloading)? 2) if a user chooses the cloud computing, how can the user choose a proper channel in order to achieve high wireless access efficiency for computation offloading?

We adopt a game theoretic approach to address these challenges. Game theory is a powerful tool for designing distributed mechanisms, such that the mobile device users in the system can locally make decisions based on strategic interactions and achieve a mutually satisfactory computation offloading solution. This can help to ease the heavy burden of complex centralized management (e.g., massive information collection from mobile device users) by the telecom cloud operator. Moreover, as different mobile devices are usually owned by different individuals and they may pursue different interests, game theory provides a useful framework to analyze the interactions among multiple mobile device users who act in their own interests and devise incentive compatible computation offloading mechanisms such that no mobile user has the incentive to deviate unilaterally.

Specifically, we model the computation offloading decision making problem among multiple mobile device users for mobile-edge cloud computing in a multi-channel wireless environment as a multi-user computation offloading game. We then propose a distributed computation offloading algorithm that can achieve the Nash equilibrium of the game. The main results and contributions of this paper are as follows:

- *Multi-User Computation Offloading Game Formulation*: We first show that it is NP-hard to find the centralized optimal multi-user computation offloading solutions in a multi-channel wireless interference environment. We hence consider the distributed alternative and formulate the distributed computation offloading decision making problem among the mobile device users as a multi-user computation offloading game, which takes into account both communication and computation aspects of mobile-edge cloud computing. We also extend our study to the scenario of multi-user computation offloading in the multi-channel wireless contention environment.
- *Analysis of Computation Offloading Game Properties*: We then study the structural property of the multi-user computation offloading game and show that the game is a potential game by carefully constructing a potential function. According to the property of potential game, we show that the multi-user computation offloading game admits the finite improvement property and always possesses a Nash equilibrium.
- *Distributed Computation Offloading Algorithm Design*: We next devise a distributed computation offloading algorithm that achieves a Nash equilibrium of the multi-user computation offloading game and derive the upper bound of the convergence time under mild conditions. We further quantify the efficiency ratio of the Nash equilibrium solution by the algorithm over the centralized optimal solutions in terms of two important metrics of the number of beneficial cloud computing users and the system-wide computation overhead. Numerical results demonstrate that the proposed algorithm can achieve efficient computation offloading performance and scale well as the user size increases.

The rest of the paper is organized as follows. We first present the system model in Section II. We then propose the multi-user computation offloading game and develop the distributed computation offloading algorithm in Sections III and IV, respectively. We next analyze the performance of the algorithm and present the numerical results in Sections V and VII, respectively. We further extend our study to the case under the wireless contention model in Section VI, discuss the related work in Section VIII, and finally conclude in Section IX.

II. System Model

We first introduce the system model. We consider a set of $\mathcal{N} = \{1, 2, \ldots, N\}$ collocated mobile device users, where each user has a computationally intensive task to be completed. There exists a wireless base-station s and through which the mobile device users can offload the computation to the cloud in proximity deployed by the telecom operator. Similar to many previous studies in mobile cloud computing (e.g., [9]–[17]) and mobile networking (e.g., [18] and [19]), to enable tractable analysis and get useful insights, we consider a quasi-static scenario where the set of mobile device users $\mathcal{N}$ remains unchanged during a computation offloading period (e.g., several hundred milliseconds), while may change across different periods.[1] Since both the communication and computation aspects play a key role in

[1]The general case that mobile users may depart and leave dynamically within a computation offloading period will be considered in a future work.

mobile-edge cloud computing, we next introduce the communication and computation models in details.

A. Communication Model

We first introduce the communication model for wireless access in mobile-edge cloud computing. Here the wireless base-station s can be a 3G/4G macro-cell or small-cell base-station [20] that manages the uplink/downlink communications of mobile device users. There are M wireless channels and the set of channels is denoted as $\mathcal{M} = \{1, 2, \ldots, M\}$. Furthermore, we denote $a_n \in \{0\} \cup \mathcal{M}$ as the computation offloading decision of mobile device user n. Specifically, we have $a_n > 0$ if user n chooses to offload the computation to the cloud via a wireless channel a_n; we have $a_n = 0$ if user n decides to compute its task locally on its own mobile device. Given the decision profile $\boldsymbol{a} = (a_1, a_2, \ldots, a_N)$ of all the mobile device users, we can compute the uplink data rate of a mobile device user n that chooses to offload the computation to the cloud via a wireless channel $a_n > 0$ as [21]

$$r_n(\boldsymbol{a}) = w \log_2 \left(1 + \frac{q_n g_{n,s}}{\varpi_0 + \sum\limits_{i \in \mathcal{N} \setminus \{n\}: a_i = a_n} q_i g_{i,s}} \right). \quad (1)$$

Here w is the channel bandwidth and q_n is user n's transmission power which is determined by the wireless base-station according to some power control algorithms such as [22] and [23].[2] Further, $g_{n,s}$ denotes the channel gain between the mobile device user n and the base-station s, and ϖ_0 denotes the background noise power. Note that here we focus on exploring the computation offloading problem under the wireless interference model, which can well capture user's time average aggregate throughput in the cellular communication scenario in which some physical layer channel access scheme (e.g., CDMA) is adopted to allow multiple users to share the same spectrum resource simultaneously and efficiently. In Section VI, we will also extend our study to the wireless contention model in which some media access control protocol such as CSMA is adopted in WiFi-alike networks.

From the communication model in (1), we see that if too many mobile device users choose to offload the computation via the same wireless access channel simultaneously during a computation offloading period, they may incur severe interference, leading to low data rates. As we discuss latter, this would negatively affect the performance of mobile-edge cloud computing.

B. Computation Model

We then introduce the computation model. We consider that each mobile device user n has a computation task $\mathcal{J}_n \triangleq (b_n, d_n)$ that can be computed either locally on the mobile device or remotely on the telecom cloud via computation offloading. Here b_n denotes the size of computation input data (e.g., the program codes and input parameters) involved in the computation task $\mathcal{J}_n$ and d_n denotes the total number of CPU cycles required to accomplish the computation task $\mathcal{J}_n$. A mobile device user n can apply the methods (e.g., call graph analysis) in [4], [24] to obtain the information of b_n and d_n. We next discuss the computation overhead in terms of both energy consumption and processing time for both local and cloud computing approaches.

1) Local Computing: For the local computing approach, a mobile device user n executes its computation task $\mathcal{J}_n$ locally on the mobile device. Let f_n^m be the computation capability (i.e., CPU cycles per second) of mobile device user n. Here we allow that different mobile devices may have different computation capabilities. The computation execution time of the task $\mathcal{J}_n$ by local computing is then given as

$$t_n^m = \frac{d_n}{f_n^m}. \quad (2)$$

For the computational energy, we have that

$$e_n^m = \gamma_n d_n, \quad (3)$$

where γ_n is the coefficient denoting the consumed energy per CPU cycle, which can be obtained by the measurement method in [15].

According to (2) and (3), we can then compute the overhead of the local computing approach in terms of computational time and energy as

$$K_n^m = \lambda_n^t t_n^m + \lambda_n^e e_n^m, \quad (4)$$

where $\lambda_n^t, \lambda_n^e \in \{0, 1\}$ denote the weighting parameters of computational time and energy for mobile device user n's decision making, respectively. When a user is at a low battery state and cares about the energy consumption, the user can set $\lambda_n^e = 1$ and $\lambda_n^t = 0$ in the decision making. When a user is running some application that is sensitive to the delay (e.g., video streaming) and hence concerns about the processing time, then the user can set $\lambda_n^e = 0$ and $\lambda_n^t = 1$ in the decision making. To provide rich modeling flexibility, our model can also apply to the generalized case where $\lambda_n^t, \lambda_n^e \in [0, 1]$ such that a user can take both computational time and energy into the decision making at the same time. In practice the proper weights that capture a user's valuations on computational energy and time can be determined by applying the multi-attribute utility approach in the multiple criteria decision making theory [25].

2) Cloud Computing: For the cloud computing approach, a mobile device user n will offload its computation task $\mathcal{J}_n$ to the cloud in proximity deployed by telecom operator via wireless access and the cloud will execute the computation task on behalf of the mobile device user.

For the computation offloading, a mobile device user n would incur the extra overhead in terms of time and energy for transmitting the computation input data to the cloud via wireless access. According to the communication model in Section II-A, we can compute the transmission time and energy of mobile device user n for offloading the input data of size b_n as, respectively,

$$t_{n,off}^c(\boldsymbol{a}) = \frac{b_n}{r_n(\boldsymbol{a})}, \quad (5)$$

[2]To be compatible with existing wireless systems, in this paper we consider that the power is determined to satisfy the requirements of wireless transmission (e.g., the specified SINR threshold). For the future work, we will study the joint power control and offloading decision making problem to optimize the performance of computation offloading. This joint problem would be very challenging to solve since the offloading decision making problem alone is NP-hard as we show later.

and

$$e_n^c(\boldsymbol{a}) = \frac{q_n b_n}{r_n(\boldsymbol{a})} + L_n, \tag{6}$$

where L_n is the tail energy due to that the mobile device will continue to hold the channel for a while even after the data transmission. Such a tail phenomenon is commonly observed in 3G/4G networks [26]. After the offloading, the cloud will execute the computation task $\mathcal{J}_n$. We denote f_n^c as the computation capability (i.e., CPU cycles per second) assigned to user n by the cloud. Similar to the mobile data usage service, the cloud computing capability f_n^c is determined according to the cloud computing service contract subscribed by the mobile user n from the telecom operator. Due to the fact many telecom operators (e.g., AT&T and T-Mobile) are capable for large-scale cloud computing infrastructure investment, we consider that the cloud computing resource requirements of all users can be satisfied. The case that a small/medium telecom operator has limited cloud computing resource provision will be considered in a future work. Then the execution time of the task $\mathcal{J}_n$ of mobile device user n on the cloud can be then given as

$$t_{n,exe}^c = \frac{d_n}{f_n^c}. \tag{7}$$

According to (5), (6), and (7), we can compute the overhead of the cloud computing approach in terms of processing time and energy as

$$K_n^c(\boldsymbol{a}) = \lambda_n^t \left(t_{n,off}^c(\boldsymbol{a}) + t_{n,exe}^c\right) + \lambda_n^e e_n^c(\boldsymbol{a}). \tag{8}$$

Similar to many studies such as [11]–[14], we neglect the time overhead for the cloud to send the computation outcome back to the mobile device user, due to the fact that for many applications (e.g., face recognition), the size of the computation outcome in general is much smaller than the size of computation input data, which includes the mobile system settings, program codes and input parameters. Also, due to the fact that wireless spectrum is the most constrained resource, and higher-layer network resources are much richer and the higher-layer management can be done quickly and efficiently via high-speed wired connection and high-performance computing using powerful servers at the base-station, the wireless access efficiency at the physical layer is the bottleneck for computation offloading via wireless transmission [10]. Similar to existing studies for mobile cloud computing [9], [17], [24], we hence account for the most critical factor (i.e., wireless access at the physical layer) only.[3]

Based on the system model above, in the following sections we will develop a game theoretic approach for devising efficient multi-user computation offloading policy for the mobile-edge cloud computing.

III. Multi-User Computation Offloading Game

In this section, we consider the issue of achieving efficient multi-user computation offloading for the mobile-edge cloud computing.

[3]We can account for the high-layer factors by simply adding a processing latency term (which is typically much smaller than the wireless access) into user's time overhead function and this will not affect the analysis of the problem.

According to the communication and computation models in Section II, we see that the computation offloading decisions $\boldsymbol{a}$ among the mobile device users are coupled. If too many mobile device users simultaneously choose to offload the computation tasks to the cloud via the same wireless channel, they may incur severe interference and this would lead to a low data rate. When the data rate of a mobile device user n is low, it would consume high energy in the wireless access for offloading the computation input data to cloud and incur long transmission time as well. In this case, it would be more beneficial for the user to compute the task locally on the mobile device to avoid the long processing time and high energy consumption by the cloud computing approach. Based on this insight, we first define the concept of beneficial cloud computing.

Definition 1: Given a computation offloading decision profile $\boldsymbol{a}$, the decision a_n of user n that chooses the cloud computing approach (i.e., $a_n > 0$) is **beneficial** *if the cloud computing approach does not incur higher overhead than the local computing approach (i.e., $K_n^c(\boldsymbol{a}) \leq K_n^m$).*

The concept of beneficial cloud computing plays an important role in the mobile-edge cloud computing. On the one hand, from the user's perspective, beneficial cloud computing ensures the individual rationality, i.e., a mobile device user would not suffer performance loss by adopting the cloud computing approach. On the other hand, from the telecom operator's point of view, the larger number of users achieving beneficial cloud computing implies a higher utilization ratio of the cloud resources and a higher revenue of providing mobile-edge cloud computing service. Thus, different from traditional multi-user traffic scheduling problem, when determining the wireless access schedule for computation offloading, we need to ensure that for a user choosing cloud computing, that user must be a beneficial cloud computing user. Otherwise, the user will not follow the computation offloading schedule, since it can switch to the local computing approach to reduce the computation overhead.

A. Finding Centralized Optimum is NP-Hard

We first consider the centralized optimization problem in term of the performance metric of the total number of beneficial cloud computing users. We will further consider another important metric of the system-wide computation overhead later. Mathematically, we can model the problem as follows:

$$\begin{aligned} \max_{\boldsymbol{a}} \quad & \sum_{n\in\mathcal{N}} I_{\{a_n>0\}} \\ \text{subject to} \quad & K_n^c(\boldsymbol{a}) \leq K_n^m,\ \forall a_n > 0, n \in \mathcal{N}, \\ & a_n \in \{0,1,\ldots,M\},\ \forall n \in \mathcal{N}. \end{aligned} \tag{9}$$

Here $I_{\{A\}}$ is an indicator function with $I_{\{A\}} = 1$ if the event A is true and $I_{\{A\}} = 0$ otherwise.

Unfortunately, it turns out that the problem of finding the maximum number of beneficial cloud computing users can be extremely challenging.

Theorem 1: The problem in (9) that computes the maximum number of beneficial cloud computing users is NP-hard.

Proof: To proceed, we first introduce the maximum cardinality bin packing problem [27]: we are given N items with sizes p_i for $i \in \mathcal{N}$ and M bins of identical capacity C, and the

objective is to assign a maximum number of items to the fixed number of bins without violating the capacity constraint. Mathematically, we can formulate the problem as

$$\begin{aligned} &\max \sum_{i=1}^{N}\sum_{j=1}^{M} x_{ij} \\ &\text{subject to} \sum_{i=1}^{N} p_i x_{ij} \leq C,\ \forall j \in \mathcal{M}, \\ &\sum_{j=1}^{M} x_{ij} \leq 1,\ \forall i \in \mathcal{N}, \\ &x_{ij} \in \{0,1\},\ \forall i \in \mathcal{N}, j \in \mathcal{M}. \end{aligned} \tag{10}$$

It is known from [27] that the maximum cardinality bin packing problem above is NP-hard.

For our problem, according to Theorem 1, we know that a user n that can achieve beneficial cloud computing if and only if its received interference $\sum_{i\in\mathcal{N}\setminus\{n\}:a_i=a_n} q_i g_{i,s} \leq T_n$. Based on this, we can transform the maximum cardinality bin packing problem to a special case of our problem of finding the maximum number of beneficial cloud computing users as follows. We can regard the items and the bins in the maximum cardinality bin packing problem as the mobile device users and channels in our problem, respectively. Then the size of an item n and the capacity constraint of each bin can be given as $p_n = q_n g_{n,s}$ and $C = T_n + q_n g_{n,s}$, respectively. By this, we can ensure that as long as a user n on its assigned channel a_n achieves the beneficial cloud computing, for an item n, the total sizes of the items on its assigned bin a_n will not violate the capacity constraint C. This is due to the fact that $\sum_{i\in\mathcal{N}\setminus\{n\}:a_i=a_n} q_i g_{i,s} \leq T_n$, which implies that $\sum_{i=1}^{N} p_i x_{i,a_n} = \sum_{i\in\mathcal{N}\setminus\{n\}:a_i=a_n} q_i g_{i,s} + q_n g_{n,s} \leq C$.

Therefore, if we have an algorithm that can find the maximum number of beneficial cloud computing users, then we can also obtain the optimal solution to the maximum cardinality bin packing problem. Since the maximum cardinality bin packing problem is NP-hard, our problem is hence also NP-hard. □

The key idea of proof is to show that the maximum cardinality bin packing problem (which is known to be NP-hard [27]) can be reduced to a special case of our problem. Theorem 1 provides the major motivation for our game theoretic study, because it suggests that the centralized optimization problem is fundamentally difficult. By leveraging the intelligence of each individual mobile device user, game theory is a powerful tool for devising distributed mechanisms with low complexity, such that the users can self-organize into a mutually satisfactory solution. This can also help to ease the heavy burden of complex centralized computing and management by the cloud operator. Moreover, another key rationale of adopting the game theoretic approach is that the mobile devices are owned by different individuals and they may pursue different interests. Game theory is a useful framework to analyze the interactions among multiple mobile device users who act in their own interests and devise incentive compatible computation offloading mechanisms such that no user has the incentive to deviate unilaterally.

Besides the performance metric of the number of beneficial cloud computing users, in this paper we also consider another important metric of the system-wide computation overhead, i.e.,

$$\begin{aligned} &\min_{\boldsymbol{a}} \quad \sum_{n\in\mathcal{N}} Z_n(\boldsymbol{a}) \\ &\text{subject to} a_n \in \{0,1,\ldots,M\},\ \forall n \in \mathcal{N}. \end{aligned} \tag{11}$$

Note that the centralized optimization problem for minimizing the system-wide computation overhead is also NP-hard, since it involves a combinatorial optimization over the multi-dimensional discrete space (i.e., $\{0,1,\ldots,M\}^N$). As shown in Sections V and VII, the proposed game theoretic solution can also achieve superior performance in terms of the performance metric of the system-wide computation overhead.

B. Game Formulation

We then consider the distributed computation offloading decision making problem among the mobile device users. Let $a_{-n} = (a_1,\ldots,a_{n-1},a_{n+1},\ldots,a_N)$ be the computation offloading decisions by all other users except user n. Given other users' decisions a_{-n}, user n would like to select a proper decision a_n, by using either the local computing ($a_n = 0$) or the cloud computing via a wireless channel ($a_n > 0$) to minimize its computation overhead, i.e.,

$$\min_{a_n\in\mathcal{A}_n \triangleq \{0,1,\ldots,M\}} Z_n(a_n, a_{-n}), \forall n \in \mathcal{N}.$$

According to (4) and (8), we can obtain the overhead function of mobile device user n as

$$Z_n(a_n, a_{-n}) = \begin{cases} K_n^m, & \text{if } a_n = 0, \\ K_n^c(\boldsymbol{a}), & \text{if } a_n > 0. \end{cases} \tag{12}$$

We then formulate the problem above as a strategic game $\Gamma = (\mathcal{N}, \{\mathcal{A}_n\}_{n\in\mathcal{N}}, \{Z_n\}_{n\in\mathcal{N}})$, where the set of mobile device users $\mathcal{N}$ is the set of players, $\mathcal{A}_n$ is the set of strategies for player n, and the overhead function $Z_n(a_n, a_{-n})$ of each user n is the cost function to be minimized by player n. In the sequel, we call the game Γ as the multi-user computation offloading game. We now introduce the important concept of Nash equilibrium.

Definition 2: A strategy profile $\boldsymbol{a}^* = (a_1^*,\ldots,a_N^*)$ is a **Nash equilibrium** *of the multi-user computation offloading game if at the equilibrium $\boldsymbol{a}^*$, no user can further reduce its overhead by unilaterally changing its strategy, i.e.,*

$$Z_n(a_n^*, a_{-n}^*) \leq Z_n(a_n, a_{-n}^*), \forall a_n \in \mathcal{A}_n, n \in \mathcal{N}. \tag{13}$$

According to the concept of Nash equilibrium, we first have the following observation.

Corollary 1: For the multi-user computation offloading game, if a user n at Nash equilibrium $\boldsymbol{a}^*$ chooses cloud computing approach (i.e., $a_n^* > 0$), then the user n must be a beneficial cloud computing user.

This is because if a user choosing the cloud computing approach is not a beneficial cloud computing user at the equilibrium, then the user can improve its benefit by just switching to the local computing approach, which contradicts with the fact that no user can improve unilaterally at the Nash equilibrium. Furthermore, the Nash equilibrium also ensures the

nice self-stability property such that the users at the equilibrium can achieve a mutually satisfactory solution and no user has the incentive to deviate. This property is very important to the multi-user computation offloading problem, since the mobile devices are owned by different individuals and they may act in their own interests.

C. Structural Properties

We next study the existence of Nash equilibrium of the multi-user computation offloading game. To proceed, we shall resort to a powerful tool of potential game [28].

Definition 3: A game is called a potential game if it admits potential function $\Phi(\boldsymbol{a})$ such that for every $n \in \mathcal{N}$, $a_{-n} \in \prod_{i\neq n}\mathcal{A}_i$, and $a'_n, a_n \in \mathcal{A}_n$, if

$$Z_n(a'_n, a_{-n}) < Z_n(a_n, a_{-n}), \tag{14}$$

we have

$$\Phi(a'_n, a_{-n}) < \Phi(a_n, a_{-n}). \tag{15}$$

An appealing property of the potential game is that it always admits a Nash equilibrium and possesses the finite improvement property, such that any asynchronous better response update process (i.e., no more than one player updates the strategy to reduce the overhead at any given time) must be finite and leads to a Nash equilibrium [28].

To show the multi-user computation offloading game is a potential game, we first show the following result.

Lemma 1: Given a computation offloading decision profile $\boldsymbol{a}$, a user n achieves beneficial cloud computing if its received interference $\mu_n(\boldsymbol{a}) \triangleq \sum_{i\in\mathcal{N}\setminus\{n\}:a_i=a_n} q_i g_{i,s}$ on the chosen wireless channel $a_n > 0$ satisfies that $\mu_n(\boldsymbol{a}) \leq T_n$, with the threshold

$$T_n = \frac{q_n g_{n,s}}{\frac{(\lambda_n^t+\lambda_n^e q_n)b_n}{2^{w\left(\lambda_n^t e_n^m+\lambda_n^e e_n^m-\lambda_n^e L_n-\lambda_n^t t_{n,exe}^c\right)}-1}} - \varpi_0.$$

Proof: According to (4), (8), and Definition 1, we know that the condition $K_n^c(\boldsymbol{a}) \leq K_n^m$ is equivalent to

$$\frac{(\lambda_n^t+\lambda_n^e q_n)\, b_n}{r_n(\boldsymbol{a})} + \lambda_n^e L_n + \lambda_n^t t_{n,exe}^c \leq \lambda_n^t t_n^m + \lambda_n^e e_n^m.$$

That is,

$$r_n(\boldsymbol{a}) \geq \frac{(\lambda_n^t+\lambda_n^e q_n)\, b_n}{\lambda_n^t t_n^m + \lambda_n^e e_n^m - \lambda_n^e L_n - \lambda_n^t t_{n,exe}^c}.$$

According to (1), we then have that

$$\sum_{i\in\mathcal{N}\setminus\{n\}:a_i=a_n} q_i g_{i,s} \leq \frac{q_n g_{n,s}}{\frac{(\lambda_n^t+\lambda_n^e q_n)b_n}{2^{w\left(\lambda_n^t e_n^m+\lambda_n^e e_n^m-\lambda_n^e L_n-\lambda_n^t t_{n,exe}^c\right)}-1}} - \varpi_0.$$

□

According to Lemma 1, we see that when the received interference $\mu_n(\boldsymbol{a})$ of user n on a wireless channel is lower enough, it is beneficial for the user to adopt cloud computing approach and offload the computation to the cloud. Otherwise, the user n should compute the task on the mobile device locally. Based on Lemma 1, we show that the multi-user computation offloading game is indeed a potential game by constructing the potential function as

$$\Phi(\boldsymbol{a}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j\neq i} q_i g_{i,s} q_j q_{j,s} I_{\{a_i=a_j\}} I_{\{a_i>0\}} + \sum_{i=1}^{N} q_i g_{i,s} T_i I_{\{a_i=0\}}. \tag{16}$$

Theorem 2: The multi-user computation offloading game is a potential game with the potential function as given in (16), and hence always has a Nash equilibrium and the finite improvement property.

Proof: Suppose that a user $k \in \mathcal{N}$ updates its current decision a_k to the decision a'_k and this leads to a decrease in its overhead function, i.e., $Z_k(a_k, a_{-k}) > Z_k(a'_k, a_{-k})$. According to the definition of potential game, we will show that this also leads to a decrease in the potential function, i.e., $\Phi(a_k, a_{-k}) > \Phi(a'_k, a_{-k})$. We will consider the following three cases: 1) $a_k > 0$ and $a'_k > 0$; 2) $a_k = 0$ and $a'_k > 0$; 3) $a_k > 0$ and $a'_k = 0$.

For case 1), since the function of $w\log_2(x)$ is monotonously increasing in terms of x, according to (1), we know that the condition $Z_k(a_k, a_{-k}) > Z_k(a'_k, a_{-k})$ implies that

$$\sum_{i\in\mathcal{N}\setminus\{k\}:a_i=a_k} q_i g_{i,s} > \sum_{i\in\mathcal{N}\setminus\{k\}:a_i=a'_k} q_i g_{i,s}. \tag{17}$$

Since $a_k > 0$ and $a'_k > 0$, according to (16) and (17), we then know that

$$\begin{aligned}
&\Phi(a_k, a_{-k}) - \Phi(a'_k, a_{-k}) \\
&= \frac{1}{2} q_k g_{k,s} \sum_{i\neq k} q_i g_{i,s} I_{\{a_i=a_k\}} \\
&\quad + \frac{1}{2}\sum_{k\neq i} q_i g_{i,s} I_{\{a_k=a_i\}} q_k g_{k,s} \\
&\quad - \frac{1}{2} q_k g_{k,s} \sum_{i\neq k} q_i g_{i,s} I_{\{a_i=a'_k\}} \\
&\quad - \frac{1}{2}\sum_{k\neq i} q_i g_{i,s} I_{\{a'_k=a_i\}} q_k g_{k,s} \\
&= q_k g_{k,s} \sum_{i\neq k} q_i g_{i,s} I_{\{a_i=a_k\}} - q_k g_{k,s} \sum_{i\neq k} q_i g_{i,s} I_{\{a_i=a'_k\}} > 0.
\end{aligned} \tag{18}$$

For case 2), since $a_k = 0$, $a'_k > 0$, and $Z_k(a_k, a_{-k}) > Z_k(a'_k, a_{-k})$, we know that $\sum_{i\in\mathcal{N}\setminus\{k\}:a_i=a'_k} q_i g_{i,s} < T_k$. This implies that

$$\begin{aligned}
&\Phi(a_k, a_{-k}) - \Phi(a'_k, a_{-k}) \\
&= q_k g_{k,s} T_k \\
&\quad - \frac{1}{2} q_k g_{k,s} \sum_{i\neq k} q_i g_{i,s} I_{\{a_i=a'_k\}} - \frac{1}{2}\sum_{k\neq i} q_i g_{i,s} I_{\{a'_k=a_i\}} q_k g_{k,s} \\
&= q_k g_{k,s} T_k - q_k g_{k,s} \sum_{i\neq k} q_i g_{i,s} I_{\{a_i=a'_k\}} > 0.
\end{aligned} \tag{19}$$

For case 3), by the similar argument in case 2), when $a_k > 0$ and $a'_k = 0$, we can also show that $Z_k(a_k, a_{-k}) > Z_k(a'_k, a_{-k})$ implies $\Phi(a_k, a_{-k}) > \Phi(a'_k, a_{-k})$.

Algorithm 1 Distributed Computation Offloading Algorithm

initialization:
each mobile device user n **chooses** the computation decision $a_n(0) = 0$.
end initialization
repeat for each user n and each decision slot t in parallel:
  transmit the pilot signal on the chosen channel $a_n(t)$ to the wireless base-station s.
  receive the information of the received powers on all the channels from the wireless base-station s.
  compute the best response set $\Delta_n(t)$.
  if $\Delta_n(t) \neq \varnothing$ **then**
    send RTU message to the cloud for contending for the decision update opportunity.
    if receive the UP message from the cloud **then**
      choose the decision $a_n(t+1) \in \Delta_n(t)$ for next slot.
    else choose the original decision $a_n(t+1) = a_n(t)$ for next slot.
    end if
  else choose the original decision $a_n(t+1) = a_n(t)$ for next slot.
  end if
until END message is received from the cloud

Combining results in the three cases above, we can hence conclude that the multi-user computation offloading game is a potential game. □

The key idea of the proof is to show that when a user $k \in \mathcal{N}$ updates its current decision a_k to a better decision a'_k, the decrease in its overhead function will lead to the decrease in the potential function of the multi-user computation offloading game. Theorem 2 implies that any asynchronous better response update process is guaranteed to reach a Nash equilibrium within a finite number of iterations. We shall exploit such finite improvement property for the distributed computation offloading algorithm design in following Section IV.

IV. Distributed Computation Offloading Algorithm

In this section we develop a distributed computation offloading algorithm in Algorithm 1 for achieving the Nash equilibrium of the multi-user computation offloading game.

A. Algorithm Design

The motivation of using the distributed computation offloading algorithm is to enable mobile device users to achieve a mutually satisfactory decision making, prior to the computation task execution. The key idea of the algorithm design is to utilize the finite improvement property of the multi-user computation offloading game and let one mobile device user improve its computation offloading decision at a time. Specifically, by using the clock signal from the wireless base-station for synchronization, we consider a slotted time structure for the computation offloading decision update. Each decision slot t consists the following two stages:

1) Wireless Interference Measurement: at this stage, we measure the interference on different channels for wireless access. Specifically, each mobile device user n who selects decision $a_n(t) > 0$ (i.e., cloud computing approach) at the current decision slot will transmit some pilot signal on its chosen channel $a_n(t)$ to the wireless base-station s. The wireless base-station then measures the total received power $\rho_m(\boldsymbol{a}(t)) \triangleq \sum_{i \in \mathcal{N}: a_i(t)=m} q_i g_{i,s}$ on each channel $m \in \mathcal{M}$ and feedbacks the information of the received powers on all the channels (i.e., $\{\rho_m(\boldsymbol{a}(t)), m \in \mathcal{M}\}$) to the mobile device users. Accordingly, each user n can obtain its received interference $\mu_n(m, a_{-n}(t))$ from other users on each channel $m \in \mathcal{M}$ as

$$\mu_n(m, a_{-n}(t)) = \begin{cases} \rho_m(\boldsymbol{a}(t)) - q_n g_{n,s}, & \text{if } a_n(t) = m, \\ \rho_m(\boldsymbol{a}(t)), & \text{otherwise.} \end{cases}$$

That is, for its current chosen channel $a_n(t)$, user n determines the received interference by subtracting its own power from the total measured power; for other channels over which user n does not transmit the pilot signal, the received interference is equal to the total measured power.

2) Offloading Decision Update: at this stage, we exploit the finite improvement property of the multi-user computation offloading game by having one mobile device user carry out a decision update. Based on the information of the measured interferences $\{\mu_n(m, a_{-n}(t)), m \in \mathcal{M}\}$ on different channels, each mobile device user n first computes its set of best response update as

$$\Delta_n(t) \triangleq \{\tilde{a} : \tilde{a} = \arg\min_{a \in \mathcal{A}_n} Z_n(a, a_{-n}(t)) \text{ and } Z_n(\tilde{a}, a_{-n}(t)) < Z_n(a_n(t), a_{-n}(t))\}.$$

Then, if $\Delta_n(t) \neq \varnothing$ (i.e., user n can improve its decision), user n will send a request-to-update (RTU) message to the cloud to indicate that it wants to contend for the decision update opportunity. Otherwise, user n will not contend and adhere to the current decision at next decision slot, i.e., $a_n(t+1) = a_n(t)$. Next, the cloud will randomly select one user k out of the set of users who have sent the RTU messages and send the update-permission (UP) message to the user k for updating its decision for the next slot as $a_n(t+1) \in \Delta_n(t)$. For other users who do not receive the UP message from the cloud, they will not update their decisions and choose the same decisions at next slot, i.e., $a_n(t+1) = a_n(t)$.

B. Convergence Analysis

According to the finite improvement property in Theorem 2, the algorithm will converge to a Nash equilibrium of the multi-user computation offloading game within finite number of decision slots. In practice, we can implement that the computation offloading decision update process terminates when no RTU messages are received by the cloud. In this case, the cloud will broadcast the END message to all the mobile device users and each user will execute the computation task according to the decision obtained at the last decision slot by the algorithm. Due to the property of Nash equilibrium, no user has the incentive to deviate from the achieved decisions.

We then analyze the computational complexity of the distributed computation offloading algorithm. In each decision

slot, each mobile device user will in parallel execute the operations in Lines 5–15 of Algorithm 1. Since most operations only involve some basic arithmetical calculations, the dominating part is the computing of the best response update in Line 11, which involves the sorting operation over M channel measurement data and typically has a complexity of $\mathcal{O}(M \log M)$. The computational complexity in each decision slot is hence $\mathcal{O}(M \log M)$. Suppose that it takes C decision slots for the algorithm to terminate. Then the total computational complexity of the distributed computation offloading algorithm is $\mathcal{O}(CM \log M)$. Let $T_{\max} \triangleq \max_{n\in\mathcal{N}}\{T_n\}$, $Q_n \triangleq q_n g_{n,s}$, $Q_{\max} \triangleq \max_{n\in\mathcal{N}}\{Q_n\}$, and $Q_{\min} \triangleq \min_{n\in\mathcal{N}}\{Q_n\}$. For the number of decision slots C for convergence, we have the following result.

Theorem 3: When T_n and Q_n are non-negative integers for any $n \in \mathcal{N}$, the distributed computation offloading algorithm will terminate within at most $\frac{Q_{\max}^2}{2Q_{\min}}N^2 + \frac{T_{\max}Q_{\max}}{Q_{\min}}N$ decision slots, i.e., $C \leq \frac{Q_{\max}^2}{2Q_{\min}}N^2 + \frac{Q_{\max}T_{\max}}{Q_{\min}}N$.

Proof: First of all, according to (16), we know that

$$0 \leq \Phi(\boldsymbol{a}) \leq \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} Q_{\max}^2 + \sum_{i=1}^{N} Q_{\max}T_{\max} = \frac{1}{2}Q_{\max}^2 N^2 + Q_{\max}T_{\max}N. \quad (20)$$

During a decision slot, suppose that a user $k \in \mathcal{N}$ updates its current decision a_k to the decision a_k' and this leads to a decrease in its overhead function, i.e., $Z_k(a_k, a_{-k}) > Z_k(a_k', a_{-k})$. According to the definition of potential game, we will show that this also leads to a decrease in the potential function by at least $Q_{\min}$, i.e.,

$$\Phi(a_k, a_{-k}) \geq \Phi(a_k', a_{-k}) + Q_{\min}. \quad (21)$$

We will consider the following three cases: 1) $a_k > 0$ and $a_k' > 0$; 2)$a_k = 0$ and $a_k' > 0$; 3) $a_k > 0$ and $a_k' = 0$.

For case 1), according to (18) in the proof of Theorem 2, we know that

$$\Phi(a_k, a_{-k}) - \Phi(a_k', a_{-k}) = Q_k\left(\sum_{i\neq k} Q_i I_{\{a_i=a_k\}} - \sum_{i\neq k} Q_i I_{\{a_i=a_k'\}}\right) > 0. \quad (22)$$

Since Q_i are integers for any $i \in \mathcal{N}$, we know that

$$\sum_{i\neq k} Q_i I_{\{a_i=a_k\}} \geq \sum_{i\neq k} Q_i I_{\{a_i=a_k'\}} + 1.$$

Thus, according to (22), we have

$$\Phi(a_k, a_{-k}) \geq \Phi(a_k', a_{-k}) + Q_k \geq \Phi(a_k', a_{-k}) + Q_{\min}.$$

For case 2), according to (19) in the proof of Theorem 2, we know that

$$\Phi(a_k, a_{-k}) - \Phi(a_k', a_{-k}) = Q_k\left(T_k - \sum_{i\neq k} Q_i I_{\{a_i=a_k'\}}\right) > 0.$$

By the similar augment as in case 1), we have

$$\Phi(a_k, a_{-k}) \geq \Phi(a_k', a_{-k}) + Q_k \geq \Phi(a_k', a_{-k}) + Q_{\min}.$$

For case 3), by the similar argument in case 2), we can also show that $\Phi(a_k, a_{-k}) \geq \Phi(a_k', a_{-k}) + Q_{\min}$.

Thus, according to (20) and (21), we know that the algorithm will terminate by driving the potential function $\Phi(\boldsymbol{a})$ to a minimal point within at most $\frac{Q_{\max}^2}{2Q_{\min}}N^2 + \frac{Q_{\max}T_{\max}}{Q_{\min}}N$ decision slots. □

Theorem 3 shows that under mild conditions the distributed computation offloading algorithm can converge in a fast manner with at most a quadratic convergence time (i.e., upper bound). Note that in practice the transmission power and channel gain are non-negative (i.e., $q_n, g_{n,s} \geq 0$), we hence have $Q_n = \{q_n g_{n,s}\} \geq 0$. The non-negative condition of $T_n \geq 0$ ensures that a user could have the chances to achieve beneficial cloud computing (otherwise, the user should always choose the local computing). For ease of exposition, we consider that Q_n and T_n are integers, which can also provide a good approximation for the general case that Q_n and T_n could be real number. For the general case, numerical results in Section VII demonstrate that the distributed computation offloading algorithm can also converge in a fast manner with the number of decision slots for convergence increasing (almost) linearly with the number of users N. Since the time length of a slot in wireless systems is typically at time scale of microseconds (e.g., the length of a slot is around 70 microseconds in LTE system [29]), this implies that the time for the computation offloading decision update process is very short and can be neglectable, compared with the computation execution process, which is typically at the time scale of millisecond/seconds (e.g., for mobile gaming application, the execution time is typically several hundred milliseconds [30]).

V. Performance Analysis

We then analyze the performance of the distributed computation offloading algorithm. Following the definition of price of anarchy (PoA) in game theory [31], we will quantify the efficiency ratio of the worst-case Nash equilibrium over the centralized optimal solutions in terms of two important metrics: the number of beneficial cloud computing users and the system-wide computation overhead.

A. Metric I: Number of Beneficial Cloud Computing Users

We first study the PoA in terms of the metric of the number of beneficial cloud computing users in the system. Let Υ be the set of Nash equilibria of the multi-user computation offloading game and $\boldsymbol{a}^* = (a_1^*, \ldots, a_N^*)$ denote the the centralized optimal solution that maximizes the number of beneficial cloud computing users. Then the PoA is defined as

$$\text{PoA} = \frac{\min_{\boldsymbol{a}\in\Upsilon}\sum_{n\in\mathcal{N}} I_{\{a_n>0\}}}{\sum_{n\in\mathcal{N}} I_{\{a_n^*>0\}}}.$$

For the metric of the number of beneficial cloud computing users, a larger PoA implies a better performance of the multi-user computation offloading game solution. Recall that $T_{\max} \triangleq \max_{n\in\mathcal{N}}\{T_n\}$, $T_{\min} \triangleq \min_{n\in\mathcal{N}}\{T_n\}$, $Q_{\max} \triangleq \max_{n\in\mathcal{N}}\{q_n g_{n,s}\}$, and $Q_{\min} \triangleq \min_{n\in\mathcal{N}}\{q_n g_{n,s}\}$. We can show the following result.

Theorem 4: Consider the multi-user computation offloading game, where $T_n \geq 0$ for each user $n \in \mathcal{N}$. The PoA for the

metric of the number of beneficial cloud computing users satisfies that

$$1 \geq \text{PoA} \geq \frac{\left\lfloor \frac{T_{\min}}{Q_{\max}} \right\rfloor}{\left\lfloor \frac{T_{\max}}{Q_{\min}} \right\rfloor + 1}.$$

Proof: Let $\tilde{\boldsymbol{a}} \in \Upsilon$ be an arbitrary Nash equilibrium of the game. Since the centralized optimum $\boldsymbol{a}^*$ maximizes the number of beneficial cloud computing users, we hence have that $\sum_{n\in\mathcal{N}} I_{\{\tilde{a}_n>0\}} \leq \sum_{n\in\mathcal{N}} I_{\{a_n^*>0\}}$ and PoA ≤ 1. Moreover, if $\sum_{n\in\mathcal{N}} I_{\{\tilde{a}_n>0\}} = N$, we have $\sum_{n\in\mathcal{N}} I_{\{a_n^*>0\}} = N$ and PoA $= 1$. In following proof, we will focus on the case that $\sum_{n\in\mathcal{N}} I_{\{\tilde{a}_n>0\}} < N$.

First, we show that for the centralized optimum $\boldsymbol{a}^*$, we have $\sum_{n\in\mathcal{N}} I_{\{a_n^*>0\}} \leq M\left(\left\lfloor \frac{T_{\max}}{Q_{\min}} \right\rfloor + 1\right)$, where M is the number of channels. To proceed, we first denote $C_m(\boldsymbol{a}) \triangleq \sum_{i=1}^{N} I_{\{a_i=m\}}$ as the number of users on channel m for a given decision profile $\boldsymbol{a}$. Since $T_n \geq 0$, we have $K_n^c(a_n, a_{-n} = \mathbf{0}) \geq K_n^m$ for $a_n > 0$, i.e., there exists at least a user that can achieve beneficial cloud computing by letting the user choose cloud computing a_n and the other users choose local computing. This implies that for the centralized optimum $\boldsymbol{a}^*$, we have $\sum_{n\in\mathcal{N}} I_{\{a_n^*>0\}} \geq 1$. Let $C_{m^*}(\boldsymbol{a}^*) = \max_{m\in\mathcal{M}}\{C_m(\boldsymbol{a}^*)\}$, i.e., channel m^* is the one with most users. Suppose user n is on the channel m^*. Then we know that

$$\sum_{i\in\mathcal{N}\setminus\{n\}:a_i=m^*} q_i g_{i,s} \leq T_n,$$

which implies that

$$\begin{aligned}(C_{m^*}(\boldsymbol{a}^*) - 1)\, Q_{\min} &\leq \sum_{i\in\mathcal{N}\setminus\{n\}:a_i=m^*} q_i g_{i,s} \\ &\leq T_n \leq T_{\max}.\end{aligned}$$

It follows that

$$C_{m^*}(\boldsymbol{a}^*) \leq \left\lfloor \frac{T_{\max}}{Q_{\min}} \right\rfloor + 1.$$

We hence have that

$$\sum_{n\in\mathcal{N}} I_{\{a_n^*>0\}} = \sum_{m=1}^{M} C_m(\boldsymbol{a}^*) \leq M C_{m^*}(\boldsymbol{a}^*) \tag{23}$$

$$\leq M\left(\left\lfloor \frac{T_{\max}}{Q_{\min}} \right\rfloor + 1\right). \tag{24}$$

Second, for the Nash equilibrium $\tilde{\boldsymbol{a}}$, since $\sum_{n\in\mathcal{N}} I_{\{\tilde{a}_n>0\}} < N$, there exists at lease one user $\tilde{n}$ that chooses the local computing approach, i.e., $a_{\tilde{n}} = 0$. Since $\tilde{\boldsymbol{a}}$ is a Nash equilibrium, we have that user $\tilde{n}$ cannot reduce its overhead by choosing computation offloading via any channel $m \in \mathcal{M}$. We then know that

$$\sum_{i\in\mathcal{N}\setminus\{\tilde{n}\}:\tilde{a}_i=m} q_i g_{i,s} \geq T_{\tilde{n}}, \forall m \in \mathcal{M},$$

which implies that

$$\begin{aligned}C_m(\tilde{\boldsymbol{a}}) Q_{\max} &\geq \sum_{i\in\mathcal{N}\setminus\{\tilde{n}\}:\tilde{a}_i=m} q_i g_{i,s}. \\ &\geq T_{\tilde{n}} \geq T_{\min}.\end{aligned}$$

It follows that

$$C_{m^*}(\tilde{\boldsymbol{a}}) \geq \frac{T_{\min}}{Q_{\max}} \geq \left\lfloor \frac{T_{\min}}{Q_{\max}} \right\rfloor.$$

Thus, we have

$$\sum_{n\in\mathcal{N}} I_{\{\tilde{a}_n>0\}} = \sum_{m=1}^{M} C_m(\tilde{\boldsymbol{a}}) \geq M\left\lfloor \frac{T_{\min}}{Q_{\max}} \right\rfloor. \tag{25}$$

Based on (24) and (25), we can conclude that PoA $\geq \frac{\left\lfloor \frac{T_{\min}}{Q_{\max}} \right\rfloor}{\left\lfloor \frac{T_{\max}}{Q_{\min}} \right\rfloor + 1}$, which completes the proof. □

Recall that the constraint $T_n \geq 0$ ensures that some user can achieve beneficial cloud computing in the centralized optimum, and avoid the possibility of the PoA involving "division by zero". Theorem 4 implies that the worst-case performance of the Nash equilibrium will be close to the centralized optimum $\boldsymbol{a}^*$ when the gap between the best and worst users in terms of wireless access performance $q_n, g_{n,s}$ and interference tolerance threshold T_n for achieving beneficial cloud computing is not large.

B. Metric II: System-Wide Computation Overhead

We then study the PoA in terms of another metric of the total computation overhead of all the mobile device users in the system, i.e., $\sum_{n\in\mathcal{N}} Z_n(\boldsymbol{a})$. Let $\bar{\boldsymbol{a}}$ be the centralized optimal solution that minimizes the system-wide computation overhead, i.e., $\bar{\boldsymbol{a}} = \arg\min_{\boldsymbol{a}\in\prod_{n=1}^{N}\mathcal{A}_n} \sum_{n\in\mathcal{N}} Z_n(\boldsymbol{a})$. Similarly, we can define the PoA as

$$\text{PoA} = \frac{\max_{\boldsymbol{a}\in\Upsilon} \sum_{n\in\mathcal{N}} Z_n(\boldsymbol{a})}{\sum_{n\in\mathcal{N}} Z_n(\bar{\boldsymbol{a}})}.$$

Note that, different from the metric of the number of beneficial cloud computing users, a smaller system-wide computation overhead is more desirable. Hence, for the metric of the system-wide computation overhead, a smaller PoA is better. Let $K_{n,min}^c \triangleq \frac{\left(\lambda_n^t + \lambda_n^e q_n\right) b_n}{w \log_2\left(1 + \frac{q_n g_{n,s}}{\varpi_0}\right)} + \lambda_n^e L_n + \lambda_n^t t_{n,exe}^c$ and

$$K_{n,max}^c \triangleq \frac{\left(\lambda_n^t + \lambda_n^e q_n\right) b_n}{w \log_2\left(1 + \frac{q_n g_{n,s}}{\varpi_0 + \left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M}\right)} + \lambda_n^e L_n + \lambda_n^t t_{n,exe}^c.$$

We can show the following result.

Theorem 5: For the multi-user computation offloading game, the PoA of the metric of the system-wide computation overhead satisfies that

$$1 \leq \text{PoA} \leq \frac{\sum_{n=1}^{N} \min\{K_n^m, K_{n,max}^c\}}{\sum_{n=1}^{N} \min\{K_n^m, K_{n,min}^c\}}.$$

Proof: Let $\tilde{\boldsymbol{a}} \in \Upsilon$ be an arbitrary Nash equilibrium of the game. Since the centralized optimum $\boldsymbol{a}^*$ minimizes the system-wide computation overhead, we hence first have that PoA ≥ 1.

For a Nash equilibrium $\hat{\boldsymbol{a}} \in \Upsilon$, if $\bar{a}_n > 0$, we shall show that the interference that a user n receives from other other users on the wireless access channel $\hat{a}_n$ is at most

$$\left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M.$$

We prove this by contradiction. Suppose that a user n at the Nash equilibrium $\hat{\boldsymbol{a}}$ receives an interference greater than $\left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M$. Then, we have that

$$\sum_{i\in\mathcal{N}\setminus\{n\}:\hat{a}_i=\hat{a}_n} q_n g_{n,s} > \left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M. \quad (26)$$

According to the property of Nash equilibrium such that no user can improve by changing the channel unilaterally, we also have that

$$\sum_{i\in\mathcal{N}\setminus\{n\}:\hat{a}_i=m} q_n g_{n,s} \geq \sum_{i\in\mathcal{N}\setminus\{n\}:\hat{a}_i=\hat{a}_n} q_n g_{n,s}, \forall m \in \mathcal{M}.$$

This implies that

$$\sum_{m=1}^{M} \sum_{i\in\mathcal{N}\setminus\{n\}:\hat{a}_i=m} q_n g_{n,s} \geq M\left(\sum_{i\in\mathcal{N}\setminus\{n\}:\hat{a}_i=\hat{a}_n} q_n g_{n,s}\right). \quad (27)$$

According to (26) and (27), we now reach a contradiction that

$$\begin{aligned}\left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M &< \sum_{i\in\mathcal{N}\setminus\{n\}:\hat{a}_i=\hat{a}_n} q_n g_{n,s} \\ &\leq \left(\sum_{m=1}^{M}\sum_{i\in\mathcal{N}\setminus\{n\}:\hat{a}_i=m} q_n g_{n,s}\right)/M \\ &\leq \left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M.\end{aligned}$$

Thus, a user n at the Nash equilibrium $\hat{\boldsymbol{a}}$ receives an interference not greater than $\left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M$. Based on this, if $\hat{a}_n > 0$, we hence have that

$$r_n(\hat{\boldsymbol{a}}) \geq w\log_2\left(1+\frac{q_n g_{n,s}}{\varpi_0+\left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M}\right),$$

which implies that

$$\begin{aligned}K_n^c(\hat{\boldsymbol{a}}) &= \frac{(\lambda_n^t+\lambda_n^e q_n)\,b_n}{r_n(\hat{\boldsymbol{a}})} + \lambda_n^e L_n + \lambda_n^t t_{n,exe}^c \\ &\geq \frac{(\lambda_n^t+\lambda_n^e q_n)\,b_n}{w\log_2\left(1+\frac{q_n g_{n,s}}{\varpi_0+\left(\sum_{i\in\mathcal{N}\setminus\{n\}} q_i g_{i,s}\right)/M}\right)} \\ &\quad + \lambda_n^e L_n + \lambda_n^t t_{n,exe}^c \\ &= K_{n,max}^c.\end{aligned}$$

Moreover, if $K_n^m < K_{n,max}^c$ and $\hat{a}_n > 0$, then the user can always improve by switching to the local computing approach (i.e., $\hat{a}_n = 0$), we thus know that

$$Z_n(\hat{\boldsymbol{a}}) \leq \min\{K_n^m, K_{n,max}^c\}. \quad (28)$$

For the centralized optimal solution $\bar{\boldsymbol{a}}$, if $\bar{a}_n > 0$, we have that

$$\begin{aligned}r_n(\bar{\boldsymbol{a}}) &= w\log_2\left(1+\frac{q_n g_{n,s}}{\varpi_0+\sum_{i\in\mathcal{N}\setminus\{n\}:\bar{a}_i=\bar{a}_n} q_i g_{i,s}}\right) \\ &\leq w\log_2\left(1+\frac{q_n g_{n,s}}{\varpi_0}\right),\end{aligned}$$

which implies that

$$\begin{aligned}K_n^c(\bar{\boldsymbol{a}}) &= \frac{(\lambda_n^t+\lambda_n^e q_n)\,b_n}{r_n(\bar{\boldsymbol{a}})} + \lambda_n^e L_n + \lambda_n^t t_{n,exe}^c \\ &\leq \frac{(\lambda_n^t+\lambda_n^e q_n)\,b_n}{w\log_2\left(1+\frac{q_n g_{n,s}}{\varpi_0}\right)} + \lambda_n^e L_n + \lambda_n^t t_{n,exe}^c \\ &= K_{n,min}^c.\end{aligned}$$

Moreover, if $K_n^m < K_{n,min}^c$ and $\bar{a}_n > 0$, then the system-wide computation overhead can be further reduced by letting user n switch to the local computing approach (i.e., $\bar{a}_n = 0$). This is because such a switching will not increase extra interference to other users. We thus know that

$$Z_n(\bar{\boldsymbol{a}}) \leq \min\{K_n^m, K_{n,min}^c\}. \quad (29)$$

According to (28) and (29), we can conclude that

$$\begin{aligned}1 \leq \text{PoA} &= \frac{\max_{\boldsymbol{a}\in\Upsilon}\sum_{n\in\mathcal{N}} Z_n(\boldsymbol{a})}{\sum_{n\in\mathcal{N}} Z_n(\bar{\boldsymbol{a}})} \\ &\leq \frac{\sum_{n=1}^{N}\min\{K_n^m, K_{n,max}^c\}}{\sum_{n=1}^{N}\min\{K_n^m, K_{n,min}^c\}}.\end{aligned}$$

□

Intuitively, Theorem 5 indicates that when the resource for wireless access increases (i.e., the number of wireless access channels M is larger and hence $K_{n,max}^c$ is smaller), the worst-case performance of Nash equilibrium can be improved. Moreover, when users have lower cost of local computing (i.e., K_n^m is smaller), the worst-case Nash equilibrium is closer to the centralized optimum and hence the PoA is lower.

VI. Extension to Wireless Contention Model

In the previous sections above, we mainly focus on exploring the distributed computation offloading problem under the wireless interference model as given in (1). Such wireless interference model is widely adopted in literature (see [21], [32] and references therein) and can well capture user's time average aggregate throughput in the cellular communication scenario in which some physical layer channel access scheme (e.g., CDMA) is adopted to allow multiple users to share the same spectrum resource simultaneously and efficiently. In this case, the multiple access among users for the shared spectrum is carried out over the signal/symbol level (e.g., at the time scale of microseconds), rather than the packet level (e.g., at the time scale of milliseconds/seconds).

In this section, we extend our study to the wireless contention model in which the multiple access among users for the shared spectrum is carried out over the packet level. This is most relevant to the scenario that some media access control protocol such as CSMA is implemented such that users content to capture the channel for data packet transmission for a long period (e.g., hundreds of milliseconds or several seconds) in the WiFi-like networks (e.g., White-Space Network [33]). In this case, we can model a user's expected throughput for computation offloading over the chosen wireless channel $a_n > 0$ as follows

$$r_n(\boldsymbol{a}) = R_n \frac{W_n}{W_n + \sum\limits_{i \in \mathcal{N}\setminus\{n\}: a_i = a_n} W_i}, \tag{30}$$

where R_n is the data rate that user n can achieve when it can successfully gab the channel, and $W_n > 0$ denotes user's weight in the channel contention/sharing, with a larger weight W_n implying that user n is more dominant in grabbing the channel. When $W_n = 1$ for any user n, it is relevant to the equal-sharing case (e.g., round robin scheduling).

Similarly, we can apply the communication and computation models in the previous sections above to compute the overhead for both local and cloud computing approaches, and model the distributed computation offloading problem as a strategic game. For such multi-user computation offloading game under the wireless contention model, we can show that it exhibits the same structural property as the case under the wireless interference model. We can first define the received "interference" (i.e., aggregated contention weights) of user n on the chosen channel as $\mu_n(\boldsymbol{a}) = \sum_{i \in \mathcal{N}\setminus\{n\}: a_i = a_n} W_i$. Then we can show the same threshold structure for the game as follow.

Lemma 2: For the multi-user computation offloading game under the wireless contention model, a user n achieves beneficial cloud computing if its received interference $\mu_n(\boldsymbol{a})$ on the chosen channel $a_n > 0$ satisfies that $\mu_n(\boldsymbol{a}) \leq T_n$, with the threshold

$$T_n = \left(\frac{\left(\lambda_n^t t_n^m + \lambda_n^e e_n^m - \lambda_n^e L_n - \lambda_n^t t_{n,exe}^c\right) R_n}{\left(\lambda_n^t + \lambda_n^e q_n\right) b_n} - 1 \right) W_n.$$

By exploiting the threshold structure above and following the similar arguments in the proof of Theorem 2, we can also show that the multi-user computation offloading game under the wireless contention model is a potential game.

Theorem 6: The multi-user computation offloading game under the wireless contention model is a potential game under the wireless contention model with the potential function as given in (31), and hence always has a Nash equilibrium and the finite improvement property.

$$\Phi(\boldsymbol{a}) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j \neq i} W_i W_j I_{\{a_i = a_j\}} I_{\{a_i > o\}} + \sum_{i=1}^{N} W_i T_n I_{\{a_n = 0\}}. \tag{31}$$

Based on Lemma 2 and Theorem (6), we observe that the multi-user computation offloading game under the wireless contention model exhibits the same structural property as the case under the wireless interference model. Moreover, by defining $q_n g_{n,s} = W_n$, the potential function in (31) is the same as that in (16). Thus, by regarding the aggregated contention weights $\mu_n(\boldsymbol{a}) = \sum_{i \in \mathcal{N}\setminus\{n\}: a_i = a_n} W_i$ as the received interference, we can apply the distributed computation offloading algorithm in Section IV to achieve the Nash equilibrium, which possesses the same performance and convergence guarantee for the case under the wireless contention model.

VII. Numerical Results

In this section, we evaluate the proposed distributed computation offloading algorithm by numerical studies. We first consider the scenario where the wireless small-cell base-station has a coverage range of 50 m [34] and $N = 30$ mobile device users are randomly scattered over the coverage region [34]. The base-station consists of $M = 5$ channels and the channel bandwidth $w = 5$ MHz. The transmission power $q_n = 100$ mWatts and the background noise $\varpi_0 = -100$ dBm [21]. According to the wireless interference model for urban cellular radio environment [21], we set the channel gain $g_{n,s} = l_{n,s}^{-\alpha}$, where $l_{n,s}$ is the distance between mobile device user n and the wireless base-station and $\alpha = 4$ is the path loss factor.

For the computation task, we consider the face recognition application in [2], where the data size for the computation offloading $b_n = 5000$ KB and the total number of CPU cycles $d_n = 1000$ Megacycles. The CPU computational capability f_n^m of a mobile device user n is randomly assigned from the set {0.5, 0.8, 1.0} GHz to account for the heterogenous computing capability of mobile devices, and the computational capability allocated for a user n on the cloud is $f_n^c = 10$ GHz [2]. For the decision weights of each user n for both the computation time and energy, we set that $\lambda_n^t = 1 - \lambda_n^e$ and λ_n^e is randomly assigned from the set {1, 0.5, 0}. In this case, if $\lambda_n^e = 1$ ($\lambda_n^e = 0$, respectively), a user n only cares about the computation energy (computation time, respectively); if $\lambda_n^e = 0.5$, then user n cares both the computation time and energy.

We first show the dynamics of mobile device users' computation overhead $Z_n(\boldsymbol{a})$ by the proposed distributed computation offloading algorithm in Fig. 2. We see that the algorithm can converge to a stable point (i.e., Nash equilibrium of the multi-user computation offloading game). Fig. 3 shows the dynamics of the achieved number of beneficial cloud computing users by the proposed algorithm. It demonstrates that the algorithm can keep the number of beneficial cloud computing users in the system increasing and converge to an equilibrium. We further show the dynamics of the system-wide computation overhead $\sum_{n \in \mathcal{N}} Z_n(\boldsymbol{a})$ by the proposed algorithm in Fig. 4. We see that the algorithm can also keep the system-wide computation overhead decreasing and converge to an equilibrium.

We then compare the distributed computation offloading algorithm with the following solutions:

A. Local Computing by All Users

each user chooses to compute its own task locally on the mobile phone. This could correspond to the scenario that each user is risk-averse and would like to avoid any potential performance degradation due to the concurrent computation offloadings by other users.

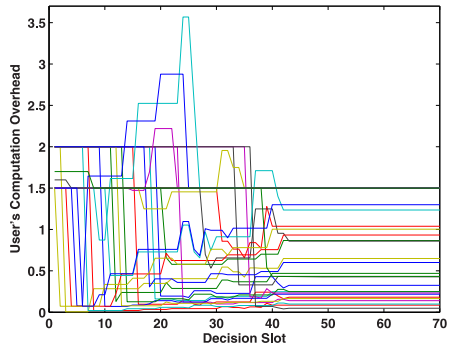


Fig. 2. Dynamics of users' computation overhead.

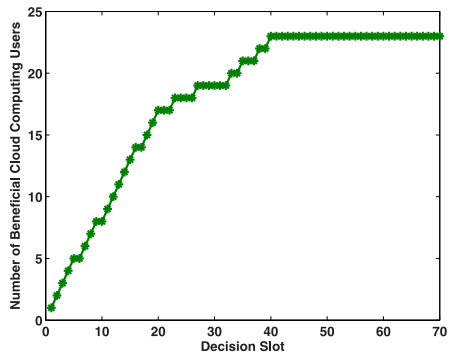


Fig. 3. Dynamics of the number of beneficial cloud computing users.

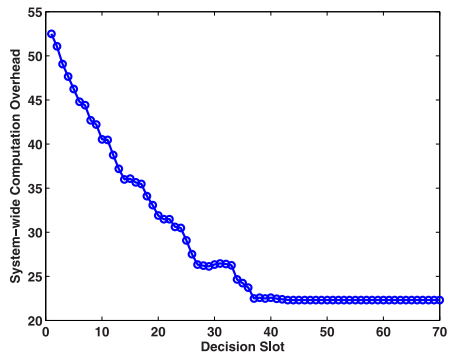


Fig. 4. Dynamics of system-wide computation overhead.

B. Cloud Computing by All Users

each user chooses to offload its own task to the cloud via a randomly selected wireless channel. This could correspond to the scenario that each user is myopic and ignores the impact of other users for cloud computing.

C. Cross Entropy Based Centralized Optimization

we compute the centralized optimum by the global optimization using Cross Entropy (CE) method, which is an advanced randomized searching technique and has been shown to be efficient in finding near-optimal solutions to complex combinatorial optimization problems [35].

We run experiments with different number of $N = 15, \ldots, 50$ mobile device users [34], respectively. We repeat each experiment 100 times for each given user number N and show the average number of beneficial cloud computing users and the average system-wide computation overhead in Figs. 5 and 6, respectively. We see that, for the metric of the number of beneficial cloud computing users, the distributed

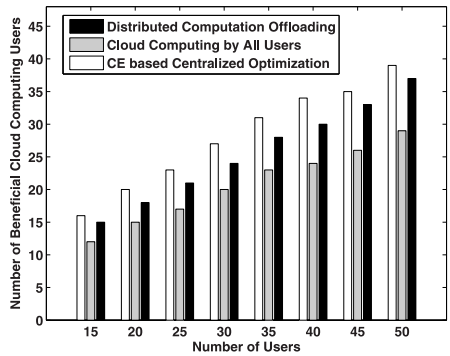


Fig. 5. Average number of beneficial cloud computing users with different number of users.

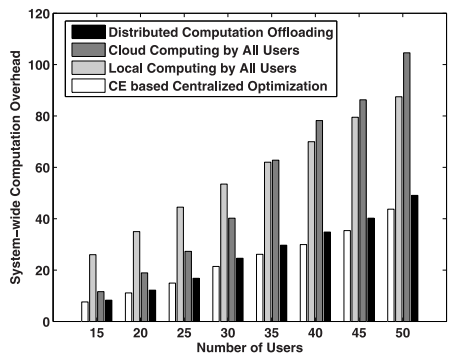


Fig. 6. Average system-wide computation overhead with different number of users.

computation offloading solution can achieve up-to 30% performance improvement over the solutions by cloud computing by all users, respectively. For the metric of the system-wide computation overhead, the distributed computation offloading solution can achieve up-to 68% and 55%, and 51% overhead reduction over with the solutions by local computing by all users, and cloud computing by all users, respectively. Moreover, compared with the centralized optimal solution by CE method, the performance loss of the distributed computation offloading solution is at most 12% and 14%, for the metrics of number of beneficial cloud computing users and system-wide computation overhead, respectively. This demonstrates the efficiency of the proposed distributed computation offloading algorithm. Note that for the distributed computation offloading algorithm, a mobile user makes the computation offloading decision locally based on its local parameters. While for CE based centralized optimization, the complete information is required and hence all the users need to report all their local parameters to the cloud. This would incur high system overhead for massive information collection and may raise the privacy issue as well. Moreover, since the mobile devices are owned by different individuals and they may pursue different interests, the users may not have the incentive to follow the centralized optimal solution. While, due to the property of Nash equilibrium, the distributed computation offloading solution can ensure the self-stability such that no user has the incentive to deviate unilaterally.

We next evaluate the convergence time of the distributed computation offloading algorithm in Fig. 7. It shows that the average number of decision slots for convergence increases

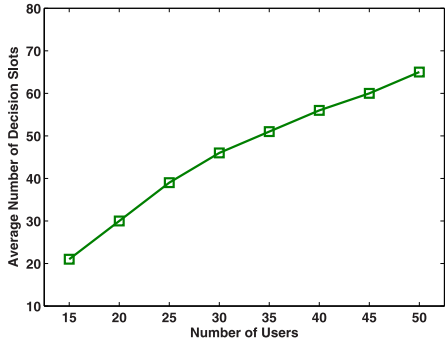


Fig. 7. Average number of decision slots for convergence with different number of users.

(almost) linearly as the number of mobile device users N increases. This demonstrates that the distributed computation offloading algorithm converges in a fast manner and scales well with the size of mobile device users in practice.[4]

VIII. Related Work

Many previous work has investigated the single-user computation offloading problem (e.g., [10]–[16]). Barbera *et al.* in [10] showed by realistic measurements that the wireless access plays a key role in affecting the performance of mobile cloud computing. Rudenko *et al.* in [11] demonstrated by experiments that significant energy can be saved by computation offloading. Gonzalo *et al.* in [12] developed an adaptive offloading algorithm based on both the execution history of applications and the current system conditions. Xian *et al.* in [13] introduced an efficient timeout scheme for computation offloading to increase the energy efficiency on mobile devices. Huang *et al.* in [14] proposed a Lyapunov optimization based dynamic offloading algorithm to improve the mobile cloud computing performance while meeting the application execution time. Wen *et al.* in [15] presented an efficient offloading policy by jointly configuring the clock frequency in the mobile device and scheduling the data transmission to minimize the energy consumption. Wu *et al.* in [16] applied the alternating renewal process to model the network availability and developed offloading decision algorithm accordingly.

To the best of our knowledge, only a few works have addressed the computation offloading problem under the setting of multiple mobile device users [9]. Yang *et al.* in [24] studied the scenario that multiple users share the wireless network bandwidth, and solved the problem of maximizing the mobile cloud computing performance by a centralized heuristic genetic algorithm. Our previous work in [17] considered the multi-user computation offloading problem in a single-channel wireless setting, such that each user has a binary decision variable (i.e., to offload or not). Given the fact that base-stations in most wireless networks are operating in the multi-channel wireless environment, in this paper we study the generalized multi-user computation offloading problem in a multi-channel setting, which results in significant differences in analysis. For example, we show the generalized problem is NP-hard, which is not true for the single-channel case. We also investigate the price of anarchy in terms of two performance metrics and show that the number of available channels can also impact the price of anarchy (e.g., Theorem 5). We further derive the upper bound of the convergence time of the computation offloading algorithm in the multi-channel environment. Barbarossa *et al.* in [9] studied the multi-user computation offloading problem in a multi-channel wireless environment, by assuming that the number of wireless access channels is greater than the number of users such that each mobile user can offload the computation via a single orthogonal channel independently without experiencing any interference from other users. In this paper we consider the more practical case that the number of wireless access channels is limited and each user mobile may experience interference from other users for computation offloading.

[4]For example, the length of a slot is at the time scale of microseconds in LTE system [29] and hence the convergence time of the proposed algorithm is very short.

IX. Conclusion

In this paper, we propose a game theoretic approach for the computation offloading decision making problem among multiple mobile device users for mobile-edge cloud computing. We formulate the problem as as a multi-user computation offloading game and show that the game always admits a Nash equilibrium. We also design a distributed computation offloading algorithm that can achieve a Nash equilibrium, derive the upper bound of convergence time, and quantify its price of anarchy. Numerical results demonstrate that the proposed algorithm achieves superior computation offloading performance and scales well as the user size increases.

For the future work, we are going to consider the more general case that mobile users may depart and leave dynamically within a computation offloading period. In this case, the user mobility patterns will play an important role in the problem formulation. Another direction is to study the joint power control and offloading decision making problem, which would be very interesting and technically challenging.

References

[1] K. Kumar and Y. Lu, "Cloud computing for mobile users: Can offloading computation save energy?," *IEEE Comput.*, vol. 43, no. 4, pp. 51–56, Apr. 2010.

[2] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman, "Cloud-vision: Real-time face recognition using a mobile-cloudlet-cloud acceleration architecture," in *Proc. IEEE ISCC*, 2012, pp. 000059–000066.

[3] J. Cohen, "Embedded speech recognition applications in mobile phones: Status, trends, and challenges," in *Proc. IEEE ICASSP*, 2008, pp. 5352–5355.

[4] E. Cuervo *et al.*, "MAUI: Making smartphones last longer with code offload," in *Proc. 8th Int. Conf. Mobile Syst., Appl., Services*, 2010, pp. 49–62.

[5] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct.–Dec. 2009.

[6] European Telecommunications Standards Institute, "Mobile-edge computing—Introductory technical white paper," 2014.

[7] U. Drolia *et al.*, "The case for mobile edge-clouds," in *Proc. 10th IEEE Int. Conf. Ubiquitous Intell. Comput.*, 2013, pp. 209–215.

[8] Ericsson, "The telecom cloud opportunity," Mar. 2012 [Online]. Available: http://www.ericsson.com/res/site_AU/docs/2012/ericsson_telecom_cloud_discussion_paper.pdf

[9] S. Barbarossa, S. Sardellitti, and P. D. Lorenzo, "Joint allocation of computation and communication resources in multiuser mobile cloud computing," in *Proc. IEEE Workshop SPAWC*, 2013, pp. 26–30.

[10] M. V. Barbera, S. Kosta, A. Mei, and J. Stefa, "To offload or not to offload? the bandwidth and energy costs of mobile cloud computing," in *Proc. IEEE INFOCOM*, 2013, pp. 1285–1293.

[11] A. Rudenko, P. Reiher, G. J. Popek, and G. H. Kuenning, "Saving portable computer battery power through remote process execution," *Mobile Comput. Commun. Rev.*, vol. 2, no. 1, pp. 19–26, 1998.

[12] G. Huertacanepa and D. Lee, "An adaptable application offloading scheme based on application behavior," in *Proc. 22nd Int. Conf. Adv. Inf. Netw. Appl.–Workshops*, 2008, pp. 387–392.

[13] C. Xian, Y. Lu, and Z. Li, "Adaptive computation offloading for energy conservation on battery-powered systems," in *Proc. IEEE ICDCS*, 2007, vol. 2, pp. 1–8.

[14] D. Huang, P. Wang, and D. Niyato, "A dynamic offloading algorithm for mobile computing," *IEEE Trans. Wireless Commun.*, vol. 11, no. 6, pp. 1991–1995, Jun. 2012.

[15] Y. Wen, W. Zhang, and H. Luo, "Energy-optimal mobile application execution: Taming resource-poor mobile devices with cloud clones," in *Proc. IEEE INFOCOM*, 2012, pp. 2716–2720.

[16] H. Wu, D. Huang, and S. Bouzefrane, "Making offloading decisions resistant to network unavailability for mobile cloud collaboration," in *Proc. IEEE Collaboratecom*, 2013, pp. 168–177.

[17] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, Apr. 2014.

[18] S. Wu, Y. Tseng, C. Lin, and J. Sheu, "A multi-channel MAC protocol with power control for multi-hop mobile ad hoc networks," *Comput. J.*, vol. 45, no. 1, pp. 101–110, 2002.

[19] G. Iosifidis, L. Gao, J. Huang, and L. Tassiulas, "An iterative double auction mechanism for mobile data offloading," in *Proc. IEEE WiOpt*, 2013, pp. 154–161.

[20] D. López-Pérez, X. Chu, A. V. Vasilakos, and H. Claussen, "On distributed and coordinated resource allocation for interference mitigation in self-organizing LTE networks," *IEEE/ACM Trans. Netw.*, vol. 21, no. 4, pp. 1145–1158, Aug. 2013.

[21] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[22] M. Xiao, N. B. Shroff, and E. K. Chong, "A utility-based power-control scheme in wireless cellular systems," *IEEE/ACM Trans. Netw.*, vol. 11, no. 2, pp. 210–221, Apr. 2003.

[23] M. Chiang, P. Hande, T. Lan, and C. W. Tan, "Power control in wireless cellular networks," *Found. Trends Netw.*, vol. 2, no. 4, pp. 381–533, 2008.

[24] L. Yang *et al.*, "A framework for partitioning and execution of data stream applications in mobile cloud computing," *Perform. Eval. Rev.*, vol. 40, no. 4, pp. 23–32, 2013.

[25] J. Wallenius *et al.*, "Multiple criteria decision making, multiattribute utility theory: Recent accomplishments and what lies ahead," *Manage. Sci.*, vol. 54, no. 7, pp. 1336–1349, 2008.

[26] W. Hu and G. Cao, "Quality-aware traffic offloading in wireless networks," in *Proc. ACM Mobihoc*, 2014, pp. 277–286.

[27] K.-H. Loh, B. Golden, and E. Wasil, "Solving the maximum cardinality bin packing problem with a weight annealing-based algorithm," in *Operations Research and Cyber-Infrastructure*. New York, NY, USA: Springer, 2009.

[28] D. Monderer and L. S. Shapley, "Potential games," *Games Econ. Behav.*, vol. 14, no. 1, pp. 124–143, 1996.

[29] T. Innovations, "LTE in a nutshell," White Paper, 2010.

[30] S. Dey, Y. Liu, S. Wang, and Y. Lu, "Addressing response time of cloud-based mobile applications," in *Proc. 1st Int. Workshop Mobile Cloud Comput. Netw.*, 2013, pp. 3–10.

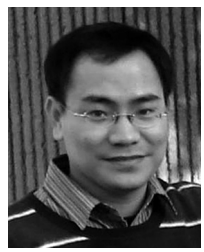
[31] T. Roughgarden, *Selfish Routing and the Price of Anarchy*. Cambridge, MA, USA: MIT Press, 2005.

[32] J. G. Andrews *et al.*, "What will 5G be?," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[33] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh, "White space networking with Wi-Fi like connectivity," *Comput. Commun. Rev.*, vol. 39, no. 4, pp. 27–38, 2009.

[34] T. Q. Quek, G. de la Roche, I. Güvenç, and M. Kountouris, *Small Cell Networks: Deployment, PHY Techniques, and Resource Management*. Cambridge, U.K.: Cambridge Univ. Press, 2013.

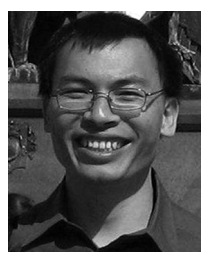
[35] R. Y. Rubinstein and D. P. Kroese, *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning*. New York, NY, USA: Springer, 2004.



Xu Chen received the Ph.D. degree in information engineering from the Chinese University of Hong Kong (Hong Kong, China) in 2012, and worked as a Postdoctoral Research Associate at Arizona State University, Tempe, USA from 2012 to 2014. He is currently a Humboldt Scholar Fellow at Institute of Computer Science of University of Göttingen, Germany. He serves as an Associate Editor of EURASIP Journal on Wireless Communications and Networking, the guest editor of International Journal of Big Data Intelligence, the special track co-chair of International Symposium on Visual Computing (ISCV'15), the publicity co-chair of International Conference on Network Games, Control and Optimization (NETGCOOP'14), and TPC members for many conferences including MOBIHOC, GLOBECOM, ICC, and WCNC. He is also the recipient of the Honorable Mention Award (first runner-up of best paper award) in 2010 IEEE international conference on Intelligence and Security Informatics (ISI), the Best Paper Runner-up Award of 2014 IEEE International Conference on Computer Communications (INFOCOM), and 2014 Hong Kong Young Scientist Award Runner-up.

Lei Jiao received the B.Sc. and M.Sc. degrees from Northwestern Polytechnical University, Xi'an, China, in 2007 and 2010, respectively, and the Ph.D. degree from University of Göttingen, Göttingen, Germany, in 2014, all in computer science. He is now a researcher with Bell Labs, Dublin, Ireland. Prior to Ph.D. study, he was a researcher with IBM Research, Beijing, China. His research interests span networking and distributed computing, with a recent focus on performance modeling, analysis, optimization, and evaluation.

Wenzhong Li received his B.S. and Ph.D. degree from Nanjing University, China, both in computer science. He was an Alexander von Humboldt Scholar Fellow in University of Göttingen, Germany. He is now an Associate Professor in the Department of Computer Science, Nanjing University. Dr. Li's research interests include wireless networks, pervasive computing, mobile cloud computing, and social networks. He has published over 40 peer-review papers at international conferences and journals, which include INFOCOM, ICDCS, IWQoS, ICPP, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, etc. He is a member of IEEE, ACM, and China Computer Federation (CCF). He was also the winner of the Best Paper Award of ICC 2009.



Xiaoming Fu received the Ph.D. degree from Tsinghua University, Beijing, China. He was a research staff at the Technical University Berlin until joining the University of Göttingen, Germany in 2002, where he has been a Professor in computer science and heading the Computer Networks Group since 2007. His research interests include network architectures, protocols, and applications. He is currently an Editorial Board member of *IEEE Communications Magazine*, IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, Elsevier *Computer Networks*, and *Computer Communications*, and has published more than 100 papers in journals and international conference proceedings. He is the coordinator of EU FP7, GreenICN, and MobileCloud projects, and received ACM ICN 2014 Best Paper Award, IEEE LANMAN 2013 Best Paper Award and the 2005 University of Göttingen Foundation Award for Exceptional Publications by Young Scholars. He is a senior member of the IEEE.