This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access

Joint Computing Resource, Power and Channel Allocations for D2D-Assisted and NOMA-based Mobile Edge Computing

Xianbang Diao, Jianchao Zheng, Yuan Wu, Yueming Cai

***Abstract*—Mobile edge computing (MEC) and non-orthogonal multiple access (NOMA) have been considered as promising techniques to address the explosively growing computation-intensive applications and accomplish the requirement of massive connectivity in the fifth-generation networks, respectively. Moreover, since the computing resources of the edge server are limited, the computing load of the edge server needs to be effectively alleviated. In this paper, by exploiting device-to-device (D2D) communication for enabling user collaboration and reducing the edge server's load, we investigate the D2D-assisted and NOMA-based MEC system. In order to minimize the weight sum of the energy consumption and delay of all users, we jointly optimize the computing resource, power and channel allocations. Regarding the computing resource allocation, we propose an adaptive algorithm to find the optimal solution. Regarding the power allocation, we present a novel power allocation algorithm based on the particle swarm optimization (PSO) for the single NOMA group comprised of multiple cellular users. Then, for the matching group comprised of a NOMA group and D2D pairs, we theoretically derive the interval of optimal power allocation and propose a PSO-based algorithm to solve it. Regarding the channel allocation, we propose a one-to-one matching algorithm based on Pareto improvement and swapping operations, and extend the one-to-one matching algorithm to a many-to-one matching scenario. Finally, we propose a scheduling-based joint computing resource, power and channel allocations algorithm (S-JCRPCA) to achieve the joint optimization. Simulation results show that the proposed solution can effectively reduce the weight sum of the energy consumption and delay of all users.**

***Index Terms*—Mobile edge computing (MEC), non-orthogonal multiple access (NOMA), device-to-device (D2D) communications, power allocation, and channel allocation.**

I. Introduction

With the proliferation of smart devices and mobile internet services, more and more mobile applications, such as augmented reality (AR), artificial intelligence (AI), and face recognition, are emerging and have attracted much attentions [1]-[2]. These sophisticated applications are generally computation-intensive and delay-sensitive, which, however, cannot be afforded by most mobile devices due to their limited computing resources and battery capacities [3]-[4]. As an interesting and promising solution in the fifth-generation (5G) communications, mobile edge computing (MEC) enables users to offload computing tasks to the edge server, which can mitigate the cost of devices and improve the quality of service (QoS) [5]-[8].

Moreover, since a large number of devices offload computation-intensive tasks to the edge server, there is an urgent requirement for massive connectivity. However, due to the allocation manner of communication resources (e.g., time slots and frequencies), conventional orthogonal multiple access (OMA) cannot satisfy the requirements of the number of user access (NUA) for MEC in 5G networks. As a potential and compelling technology, non-orthogonal multiple access (NOMA) enables multiple users to share one channel at the same time, thereby improving NUA and spectral efficiency (SE) [9]-[13].

Although NOMA can address the requirement of massive connectivity, it introduces a new challenge, namely, when vast users offload tasks to the edge server, the users' experienced QoS will be impaired due to the excessive computation workload at the edge server. Many approaches have been proposed to alleviate this issue. The authors of [14] utilized the repeatability of computing results of AR to store reusable results in servers to mitigate the computing load. However, this scheme often requires a specific scenario. The authors of [15]-[18] utilized the cooperation between servers or servers and clouds to balance the computing load. However, these schemes usually require neighboring servers and remote clouds to coordinate their task allocations for the load balancing, which usually result in a heavy signalling overhead in the backhaul. Thus, it is necessary to explore a new way of cooperation to mitigate the computing load of edge servers in the NOMA-based MEC system.

Device-to-device (D2D) communication has been considered as an important paradigm in 5G systems and has drawn lots of research interests [19]-[21]. In D2D-enabled cellular networks, devices are allowed to communicate directly through cellular channels, which not only mitigates the workload of the base station (BS) but also improves SE and NUA [22]-[23]. Therefore, in this work, we exploit the D2D to reduce the computing load of the edge server in the NOMA-based MEC system. In the system, cellular users (CUs) form multiple NOMA groups, and users in each NOMA group offload tasks to the edge server through the same subchannel. Since the

This work is supported by the National Natural Science Foundation of China under Grant 61801505, by the Jiangsu Provincial Nature Science Foundation of China under Grant No. BK20170755, and by the National Postdoctoral Program for Innovative Talents of China under Grant BX201700109.

X. Diao, J. Zheng, Y. Cai are with the College of Communications Engineering, Army Engineering University of PLA, Nanjing 210007, China (e-mail: diaoxianbang1994@163.com; longxingren.zjc.s@163.com; caiym@vip.sina.com). Y. Wu is with the College of Information Engineering, Zhejiang University of Technology, Hangzhou, 310023, China (email: iewuy@zjut.edu.cn). J. Zheng is also with the National Innovation Institute of Defense Technology, Academy of Military Sciences of PLA, Beijing 100010, China.

computing resource of different devices is heterogeneous, the device with weak computing capability can offload task to the device with stronger computing capability through a cellular channel. However, for the D2D-assisted and NOMA-based MEC, there are still many factors (such as the multi-user interference and resource allocation) that affect the system performance (e.g., energy consumption and delay). Therefore, interference management and resource allocation are essential for the D2D-assisted and NOMA-based MEC system.

Recently, several studies have investigated the performance of NOMA-based MEC system. The authors of [24] considered the sociality and cooperation between devices, and optimized the system delay under the constraints of energy consumption and power consumption. The authors of [25] considered a multi-antenna NOMA-based MEC system and optimized the energy efficiency with the constraints of the rate and power. In [26], multiple users offload tasks to unmanned aerial vehicles (UAVs) in the way of NOMA and reduced the energy consumption of system by optimizing the power of all devices and the trajectory of UAVs. The authors of [27] exploited NOMA into MEC and formulated an optimization framework to reduce the energy consumption of system by optimizing user clustering, resource and power allocations. Moreover, some studies have focused on the D2D-based MEC system. For instance, the authors of [28] proposed a novel D2D crowd framework for the MEC system to achieve energy-efficient collaborative task executions at the network edge for mobile users.

Moreover, inspired by the benefits of NOMA and D2D, some studies investigated the combination of the two technologies for further improving the SE and NUA. Thus, recently, there have been some studies investigated the D2D communication underlaying a NOMA-based cellular network [29]-[32]. The authors of [29] considered a novel NOMA enhanced D2D communication scheme and proposed a novel algorithm based on the matching theory and sequential convex programming to maximize the system sum rate. The authors of [30] investigated the resource allocation problem in the D2D communication underlaying a NOMA-based energy-harvesting cellular network. Focusing on a uplink multi-carrier NOMA (MC-NOMA) in D2D underlaying cellular networks, the authors of [31] proposed an iterative algorithm that applies the Karush-Kuhn-Tucker (KKT) conditions to solve the power allocation problem.

The above studies have investigated the NOMA or D2D based MEC system and D2D communication underlaying NOMA-based cellular network. However, for interference management, some studies focused on one aspect (either power control or channel allocation). In [32], a joint power allocation and channel allocation was investigated by exploiting the techniques of lagrangian duality and dynamic programming. This work considered a simplified NOMA-based scenario and internal interference without considering the influence of inter-group interference. Although the authors of [33] investigated the internal and inter-group interference and proposed a joint optimization of power and channel allocations in D2D communication underlaying a NOMA-based network, this work did not consider the joint optimization for both cellular and D2D users. Moreover, many existing studies focus on the interference management, and pay little attention to the optimization of computing resources which is significant for the performance of MEC system. To the best of our knowledge, there is no existing work investigating the D2D-assisted and NOMA-based MEC system.

Therefore, in this work, we jointly optimize the computing resource, power and channel allocations to reduce the energy consumption and delay of both CUs and D2D users. In the system, all CUs will form multiple NOMA groups. In each NOMA group, CUs offload tasks to the edge server through a fixed subchannel. Each D2D pair consists of a task requester and a task agent, where the task requester offloads a task to the task agent through one arbitrary subchannel. In order to minimize the weight sum of energy consumption and delay of all users, we jointly optimize the computing resource, power and channel allocations. However, the objective function is a non-convex multivariate fractional summation function and the constraints are also non-convex. After analysis, we decouple the original problem into three subproblems. The main contributions of this work can be summarized as follows:

1) We simultaneously optimize the energy consumption and delay for cellular and D2D users under the constraints of energy consumption, delay, power and computing resource. The optimization problem is challenging due to the structure of multivariate fractional summation. We analyze the relationship between the three variables in the optimization problem, and decouple the optimization problem into three sub-problems, and obtain a close-to-optimum solution.
2) We investigate the power allocation problem with two different matching cases. First, for the single NOMA group, we propose an iterative particle swarm optimization (PSO) algorithm. In each iteration, it transforms the multivariate problem into a univariate problem to reduce the computational complexity. Secondly, for the matching group, we reduce the complexity of the power allocation problem by transforming the multivariate problem into a univariate problem by replacing the CUs' power with the task requester's power. Then, we quantify the interval of the optimal task requester's power allocation and further utilize the PSO algorithm to find the optimal task requester's power within this derived interval.
3) For channel allocation, we propose a one-to-one matching algorithm based on the swapping operations and Pareto improvement, and prove that the algorithm converges to the locally or globally optimal solution. Moreover, in order to achieve the goal of joint optimization, we propose a scheduling-based joint computing resource, power and channel allocations algorithm.

The remainder of this paper is organized as follows. Section II presents the system model. In Section III, we formulate an optimization problem and decouple it to three subproblems. In Section IV and Section V, we solve the three subproblems and propose an algorithm to solve the joint optimization problem. Numerical results are presented in Section VI. Finally, con-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access

clusion is given by Section VII.

II. System Model

We consider a cellular uplink communication system which consists of a BS equipped with a MEC server, U CUs and N D2D pairs, denoted by $\mathcal{D} = \{D_1, D_2, \cdots, D_N\}$, as illustrated in Fig. 1. In the system, the CUs and D2D pairs are uniformly distributed. According to the user clustering method in [34], CUs are divided into M NOMA groups at equal intervals of channel gains. Denote $\mathcal{NG} = \{NG_1, NG_2, \cdots, NG_M\}$ as the set of NOMA groups, where $NG_m = \{NG_m^1, NG_m^2, \cdots, NG_m^{K_m}\}$ denotes the m-th NOMA group, and NG_m^j is the j-th user in the m-th NOMA group, and K_m denotes the number of users in the m-th NOMA group. In each D2D pair, there are a sender called D2D task requester (DTR) and a receiver called D2D task agent (DTA). Moreover, we assume that DTR and DTA are friends or relatives with solid mutual trust relationship, so DTA can help DTR without rewards. Both CUs and DTRs have a computation-intensive and delay-sensitive task $A_i \triangleq (D_i, C_i, T_i^{\max})$, where D_i is the input-data size (in bits), X_i represents the computing intensity (in CPU cycles per bit), and $T_i^{\max}$ represents the deadline of each task. In addition, there are M orthogonal subchannels $\mathcal{SC} = \{SC_1, SC_2, \cdots, SC_M\}$, and we define $B = \frac{W}{M}$ as the bandwidth of each subchannel, where W is the available channel bandwidth. In particular, each NOMA group is assigned a subchannel, and CUs who belong to the same NOMA group offload tasks to the edge server through the same subchannel. DTRs can select one arbitrary subchannel for computing offloading. In other words, each subchannel may not be occupied by any D2D pair or may be occupied by multiple D2D pairs. In addition, we assume that the computing resources of the edge server are divided into multiple computing cells, and the total number of computing cells is CN. Moreover, we assume that all users perform computing offloading simultaneously and the BS has complete channel state information (CSI).

A. *Communication Model*

We assume that subchannel SC_m is occupied by the NOMA group NG_m. Since each DTR can select any subchannel to offload its task, there are two matching cases for a NOMA group. Specifically, if NOMA group NG_m does not share its subchannel SC_m with any D2D pair, then we call NOMA group NG_m as a single NOMA group. Otherwise (namely, some D2D pairs reuse the subchannel SC_m of NOMA group NG_m), we call NOMA group NG_m as a matching group. Then, the received signal at the BS corresponding to subchannel SC_m is given by

$$y_m = \sum_{j=1}^{K_m} \sqrt{p_{NG_m^j}} g_{NG_m^j,B} x_{NG_m^j} + \sum_{n=1}^{N} \alpha_{mn} \sqrt{p_n} g_{nB} e_n + \xi_m, \quad (1)$$

where $x_{NG_m^j}$ and e_n are the transmit signals of NG_m^j and DTR_n, respectively. ξ_m is the additive white Gaussian noise (AWGN) at the BS on subchannel SC_m with variance σ^2. $g_{NG_m^j,B}$ is the channel gain between NG_m^j and BS, and g_{nB} is the channel gain between DTR_n and BS. α_{mn} is a binary variable which represents the channel selection decision. When $\alpha_{mn} = 1$, the NG_m shares the same subchannel with the D_n. $p_{NG_m^j}$ and p_n are the power of NG_m^j and DTR_n, respectively. Moreover, we can obtain the received signal of the DTA_n, which is given by

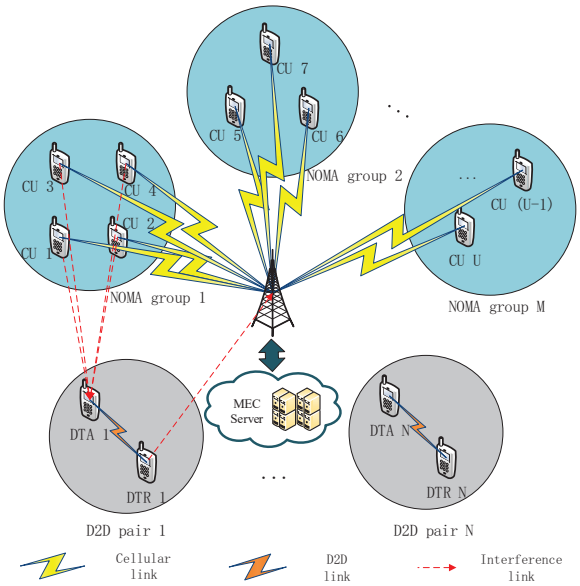


Fig. 1. System model

$$z_n = \sqrt{p_n} g_{nn} e_n + \sum_{m=1}^{M} \sum_{j=1}^{K_m} \alpha_{mn} \sqrt{p_{NG_m^j}} g_{NG_m^j,n} x_{NG_m^j} + \sum_{m=1}^{M} \sum_{l=1,l\neq n}^{N} \alpha_{mn} \alpha_{ml} \sqrt{p_l} g_{ln} e_l + \xi_n, \quad (2)$$

where g_{nn} is the channel gain between DTR_n and DTA_n, and $g_{NG_m^j,n}$ is the channel gain between NG_m^j and DTA_n. g_{ln} is the channel gain between DTR_l and DTA_n. ξ_n is the additive white Gaussian noise (AWGN) at the DTA_n with variance σ^2.

According to the sorting method of the CUs in the NOMA group in [34], uplink users will be interfered by other users' signals with lower channel gains. Thus, there exist the intra-group interference which comes from the inside of the NOMA group. In addition, there also exist other interferences which come from DTRs which share the same subchannel with the NOMA group. Hence, the signal-to-noise ratio (SINR) of NG_m^j is given by

$$\Gamma_{NG_m^j} = \frac{p_{NG_m^j} g_{NG_m^j}}{\sum_{l=j+1}^{K_m} p_{NG_m^l} g_{NG_m^l} + \sum_{n=1}^{N} \alpha_{mn} p_n g_{nB} + \sigma^2}, \quad (3)$$

where σ^2 is the background noise power. For the D2D pair, the interference comes from the NOMA group which shares the same subchannel with it. The SINR of DTR_n is given by

$$\Gamma_n = \frac{p_n g_{nn}}{\sum_{m=1}^{M} \sum_{j=1}^{K_m} \alpha_{mn} p_{NG_m^j} g_{NG_m^j,n} + \sum_{m=1}^{M} \sum_{l=1,l\neq n}^{N} \alpha_{mn} \alpha_{ml} p_l g_{ln} + \sigma^2}, \quad (4)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access

and the achievable data rates of NG_m^j and DTR_n are given by

$$R_{NG_m^j} = B\log_2(1+\Gamma_{NG_m^j}), \tag{5}$$

$$R_n = B\log_2(1+\Gamma_n). \tag{6}$$

B. Cost Model

In this work, we focus on the energy consumption and delay of the system. Similar to [35][36], in this paper, we do not consider the energy consumption of the edge server, DTAs and the process of reception. Therefore, only CUs and DTRs have energy consumption for the process of task uploading. The energy consumption of each CU (or DTR) is given by

$$E_i = p_i\frac{D_i}{R_i}, i \in \{NG_m, m\in[1,M]\}\cup\{DTR_n, n\in[1,N]\}. \tag{7}$$

Besides, the task delay consists of three parts, i.e., the task uploading delay T^{up}, the task execution delay T^{exe}, and the task result download delay T^{down}. Due to the small data size of the computing results, we do not consider the T^{down}. Thus, the task delay of CUs and DTRs is given by

$$T_i = T_i^{up}+T_i^{exe}, i \in \{NG_m, m\in[1,M]\}\cup\{DTR_n, n\in[1,N]\}, \tag{8}$$

where $T_i^{up} = \frac{D_i}{R_i}$ and $T_i^{exe} = \frac{C_i}{f_i}$. f_i is the computing rate (in the unit of CPU cycles per second) for the task of user i, and it is given by

$$f_i = \begin{cases} \beta_{NG_m^j}F_{BS}, i\in\{NG_m^j\}, m\in[1,M], j\in[1,K_m] \\ F_n^{clo}, i\in\{DTR_n\}, n\in[1,N] \end{cases}, \tag{9}$$

where F_{BS} and F_n^{clo} are the computing rates of the computing cell of the edge server and DTA_n, respectively. $\beta_{NG_m^j}$ denotes the number of computing cells assigned to the NG_m^j.

In the process of computing offloading, both energy consumption and delay are vital. Similar to [37], we introduce the non-negative weight factor ω to trade off the energy consumption and delay. Thus, the weight sum of the energy consumption and delay of all users in the system is given by

$$\begin{aligned} cost = \omega(\sum_{m=1}^{M}\sum_{j=1}^{K_m}E_{NG_m^j} + \sum_{n=1}^{N}E_n) \\ +(1-\omega)(\sum_{m=1}^{M}\sum_{j=1}^{K_m}T_{NG_m^j} + \sum_{n=1}^{N}T_n) \end{aligned}, \tag{10}$$

where $E_{NG_m^j}$ and E_n are the energy consumption of NG_m^j and DTR_n, and $T_{NG_m^j}$ and T_n are the task delay of NG_m^j and DTR_n, respectively. In addition, ω is the weight of the energy consumption and $1-\omega$ is the weight of the delay.

III. Optimization Problem

In this section, we formulate an optimization problem to minimize the weight sum of the cost of all users with constraints on the energy consumption, power, delay and computing resources. The optimization problem is given by

$$\mathbf{P1}: \min_{\alpha_{mn},\beta_{NG_m^j},p_{NG_m^j},p_n} cost, \tag{11a}$$

$$s.t. \quad E_{NG_m^j} \le E_{NG_m^j}^{\max}, \forall m\in[1,M], \forall j\in[1,K_m], \tag{11b}$$

$$E_n \le E_n^{\max}, \forall n\in[1,N], \tag{11c}$$

$$T_{NG_m^j} \le T_{NG_m^j}^{\max}, \forall m\in[1,M], \forall j\in[1,K_m], \tag{11d}$$

$$T_n \le T_n^{\max}, \forall n\in[1,N], \tag{11e}$$

$$p_i^{\min} \le p_i \le p_i^{\max}, \forall i\in\{NG_m, m\in[1,M]\} \cup\{n, n\in[1,N]\}, \tag{11f}$$

$$\alpha_{mn}\in\{0,1\}, \forall m\in[1,M], \forall n\in[1,N], \tag{11g}$$

$$\sum_{m=1}^{M}\alpha_{mn} = 1, \forall n\in[1,N], \tag{11h}$$

$$0 \le \sum_{n=1}^{N}\alpha_{mn} \le \alpha_{\max}, \forall m\in[1,M], \tag{11i}$$

$$\beta_{NG_m^j}\in N^+, \forall j\in[1,K_m], \tag{11j}$$

$$1 \le \beta_{NG_m^j} \le \beta_{\max}, \forall m\in[1,M], \forall j\in[1,K_m], \tag{11k}$$

$$\sum_{m=1}^{M}\sum_{j=1}^{K_m}\beta_{NG_m^j} = CN, \tag{11l}$$

where constraints (11b) and (11c) are energy consumption constraint conditions. Constraints (11d) and (11e) are task delay constraint conditions. Constraint (11f) is transmission power constraint condition. Constraint (11g) indicates that the α_{mn} is a binary variable. Constraint (11h) indicates that each D2D pair must share a subchannel with one NOMA group. Constraint (11i) shows that the NOMA group may not share subchannel with any D2D pair and the number of D2D pairs that share subchannel with the NOMA group shall not exceed $\alpha_{\max}$. Constraint (11j) denotes the $\beta_{NG_m^j}$ is a positive integer. Constraint (11k) denotes each CU is assigned at least one computing cell and at most $\beta_{\max}$ computing cells, where $\beta_{\max} = CN - U$. Constraint (11l) indicates that the total number of computing cells allocated to all CUs is equal to the total number of computing cells of the edge server.

$\mathbf{P1}$ is a mixed integer non-linear programming (MINLP) problem which consists of binary, integer and real variables, and the objective function is a non-convex function. There is no efficient approach to solve this problem optimally. Hence, we have an analysis of $\mathbf{P1}$ as follows. Firstly, we find that computing resource allocation variables $\beta_{NG_m^j}$ are independent of the channel allocation decision variables α_{mn} and transmission power $p_{NG_m^j}$ and p_n. Therefore, computing resource allocation can be performed separately from power and channel allocations. Secondly, the power and channel allocations are coupled, and it is difficult to solve these two problems synchronously. Similar to [29][38], we consider decoupling the power and channel allocation problems. Therefore, we decompose the original problem into three subproblems. First, we prioritize the computing resource allocation

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access

problem to obtain the optimal task execution delay. Secondly, on the premise of determining the task execution delay, we propose algorithms to solve power allocation problem for the single NOMA group and matching group. Then, we investigate the channel allocation algorithms for D2D pairs. Finally, we combine the proposed algorithms of the three subproblems to obtain the minimum weight sum.

IV. Computing Resource and Power Allocation

In this section, we first solve the computing resource allocation problem which is related to task execution delay. The types of task execution delay of different users are different. For CUs, since their task execution delay depends on the computing resource allocation strategy, it can be considered as a variable. For D2D users, since the computing rate of DTA is a constant, their task execution delay can be considered as a constant. In other words, the computing resource allocation is only for CUs. The optimization problem of computing resource allocation is given by

$$\mathbf{P2}: \min_{\beta_{NG_m^j}} (1-\omega)\Big(\sum_{m=1}^{M}\sum_{j=1}^{K_m}\frac{C_{NG_m^j}}{\beta_{NG_m^j}F_{BS}}+\sum_{n=1}^{N}\frac{C_n}{F_n^{clo}}\Big), \tag{12a}$$

$$s.t. \quad \beta_{NG_m^j} > \frac{C_{NG_m^j}}{T_{NG_m^j}F_{BS}}, \forall m\in[1,M], \forall j\in[1,K_m], \tag{12b}$$

$$Constraints(11j, 11k), \tag{12c}$$

in which constraint (12b) comes from constraint (11d) and ensures that the computing cells assigned to each CU satisfy the minimum delay requirements. Since $\sum_{n=1}^{N}\frac{C_n}{F_n^{clo}}$ is a constant, we can optimize $\mathbf{P2}$ without taking it into account. We propose a computing resource allocation algorithm (CRA) to minimize the sum of the task execution delay of CUs. The pseudo-code for the CRA is shown in Algorithm 1. As CRA shows, for each CU, we first ensure that its task execution delay is lower than the task delay. We define a task execution delay gain $gain_{NG_m^j}^{exe} = \frac{C_{NG_m^j}}{\beta_{NG_m^j}F_{BS}} - \frac{C_{NG_m^j}}{(\beta_{NG_m^j}+1)F_{BS}}$. To minimize the task execution delay, for each computing cell, we assign it to the user with the maximum task execution gain. In addition, n_c denotes the number of remaining computing cells in the allocation process.

Proposition 1 The CRA converges to the optimal computing resource allocation strategy.

Proof: It is worth mentioning that the allocation strategy of computing resource is directly related to the task execution delay of the CUs. In other words, the optimal computing resource allocation strategy corresponds to the optimal task execution delay. Thus, we investigate whether CRA can converge to the optimal task execution delay.

First, we assume that the solution of the computing resource allocation variables $\beta_{NG_m^j}$ obtained by CRA is S, and the optimal solution is S^*. Moreover, we assume $T^{exe}(S) > T^{exe}(S^*)$ where $T^{exe}(\cdot)$ is a function of the solution and represents the total task execution delays of all CUs. In the last iteration of the CRA, the user's parameters with the maximum task execution delay gain are $[m, j]$. As mentioned above, due to $T^{exe}(S) > T^{exe}(S^*)$, there must be $[m^*, j^*] \neq [m, j]$ to make $gain_{NG_{m^*}^{j^*}}^{exe} > gain_{NG_m^j}^{exe}$. This is contrary to the rule of CRA. Therefore, the optimal solution of task execution delay is obtained by CRA. In summary, the CRA can converge to the optimal computing resource allocation strategy. ■

Algorithm 1 Computing Resource Allocation Algorithm (CRA)

1: **Initialization:** $\beta_{NG_m^j}=0, \forall m\in[1,M], \forall j\in[1,K_m]; n_c = CN$
2: **for all** $m\in[1,M]$ **do**
3: **for all** $j\in[1,K_m]$ **do**
4: **repeat**
5: $T_{NG_m^j}^{exe} = \frac{C_{NG_m^j}}{(\beta_{NG_m^j}+1)F_{BS}}, n_c = n_c - 1$; $\beta_{NG_m^j} = \beta_{NG_m^j} + 1$
6: **until** $T_{NG_m^j}^{exe} < T_{NG_m^j}$
7: **end for**
8: **end for**
9: **repeat**
10: Calculate $gain_{NG_m^j}^{exe}, \forall m\in[1,M], \forall j\in[1,K_m]$.
11: $[m^*, j^*] = \underset{m,j}{argmax}(\{gain_{NG_m^j}^{exe}\})$.
12: $\beta_{NG_{m^*}^{j^*}} = \beta_{NG_{m^*}^{j^*}} + 1$.
13: $n_c = n_c - 1$.
14: **until** $n_c = 0$

Proposition 2 The computational complexity of CRA is $\mathcal{O}(CN)$.

Proof: In CRA, the process of computing resource allocation is divided into two parts. The first part is to ensure that minimum delay requirements are satisfied and the second part is to assign remaining computing cells to the user with the maximum task execution gain in each iteration. At the same time, the computing complexity of each iteration in each part is same and the complexity between the two parts is almost same. Therefore, we can consider the complexity of each iteration as a constant. It can be seen that the total number of iterations is CN, meaning that the computing complexity of CRA is $\mathcal{O}(CN)$. ■

A. Power Allocation of Single NOMA Group

Based on the results of the computing resource allocation, we investigate the power allocation problem for single NOMA group and matching group, respectively. We assume that NG_m is a single NOMA group. Thus, the power allocation problem of the single NOMA group is given by

$$\mathbf{P3}: \min_{\{p_{NG_m^j}, j\in[1,K_m]\}} \sum_{j=1}^{K_m}\Big(\omega p_{NG_m^j}\frac{D_{NG_m^j}}{R_{NG_m^j}} + (1-\omega)\Big(\frac{D_{NG_m^j}}{R_{NG_m^j}} + T_{NG_m^j}^{exe}\Big)\Big), \tag{14a}$$

$$s.t. \quad E_{NG_m^j} \le E_{NG_m^j}^{\max}, \forall j\in[1,K_m], \tag{14b}$$

$$T_{NG_m^j} \le T_{NG_m^j}^{\max}, \forall j\in[1,K_m], \tag{14c}$$

$$p_{NG_m^j}^{\min} \le p_{NG_m^j} \le p_{NG_m^j}^{\max}, \forall j\in[1,K_m], \tag{14d}$$

where $T^{exe}_{NG^j_m}$ is a constant. Since NG_m is a single NOMA group, $\Gamma_{NG^j_m} = \frac{p_{NG^j_m} g_{NG^j_m}}{\sum_{l=j+1}^{K_m} p_{NG^l_m} g_{NG^l_m} + \sigma^2}$. However, $\mathbf{P3}$ is a multivariate nonlinear fractional summation problem. To tackle this problem, based on the PSO, we propose a power allocation algorithm for single NOMA group (PASNG). The algorithm is summarized in Algorithm 2, where rn is the number of rounds, and K_m is the period. sn is the serial number of the current optimized user. N_{pop} is the number of PSO populations. N_{iter} is the number of iterations of the PSO. $p^*_{NG^j_m,rn}$ is the power of NG^j_m at the end of the rounds rn, and ε_p is the power error threshold. As shown in PASNG, the main idea of the algorithm can be summarized as the following two points. First, in order to avoid high computing complexity, only one user's power is updated in each iteration, and the power of other users are kept unchanged. This process can transform a multivariate problem into a single variable problem, and reduce the computing complexity as well as the probability that the PSO algorithm falls into a locally optimal solution. Secondly, in each iteration, we utilize the PSO to optimize the power of the corresponding user. When power of all users are optimized, the above operation is performed again until the convergence conditions are met, where the convergence condition indicates that the power of all CUs in the NOMA group is almost no longer changed. Furthermore, similar to [39], we do not consider the energy consumption generated by the implementation of the PSO algorithm.

Algorithm 2 Power Allocation Algorithm Based on PSO for Single NOMA Group (PASNG)

1: **Initialization:** $rn = 1, p_{NG^j_m} = p^{\max}_{NG^j_m}, \forall j \in [1, K_m], sn = K_m, N_{pop} = 20, N_{iter} = 50$
2: **repeat:**
3: According to the results of initialization or last iteration, fix users' power other than the sn user.
4: Use PSO to optimize the power of the sn user.
5: $sn = sn - 1$.
6: **if** $sn = 1$ **then**
7: $sn = K_m, rn = rn + 1$.
8: **end if**
9: **until** $\frac{1}{K_m}\sum_{j=1}^{K_m} |p^*_{NG^j_m,rn} - p^*_{NG^j_m,rn-1}| < \varepsilon_p$
10: **Output:** $\{p^{s,*}_{NG^j_m}, j \in [1, K_m]\}$

Proposition 3 The PASNG converges to the locally or globally optimal solution.

Proof: Similar to [39], we set the parameters of the PSO to satisfy the condition of converging to a locally or globally optimal solution. This allows each iteration to obtain a locally or globally optimal solution under current power conditions. Moreover, we assume that the power results of the first iteration is $\{p^1_{NG^j_m}, j \in [1, K_m]\}$. During the second iteration, the next user's power will be optimized. It is worth noting that this optimization is carried out on the basis of power result of the first iteration. Therefore, the result of the second iteration is better than the result of the first iteration, and so on. It can be seen from the above analysis that as the number of iterations increases, the cost will continue to decrease. When each user's transmit-power does not change between two consecutive iterations, the locally or globally optimal solution is obtained. ■

Proposition 4 The computational complexity of PASNG is $\mathcal{O}(K^2_m \times N_{pop} \times N_{iter})$.

Proof: PASNG adopts the PSO in each iteration. The computing complexity of PSO is related to the population number and total number of iterations. Similar to [39], the computational complexity of each iteration should be $\mathcal{O}(N_{pop} \times N_{iter})$. In addition, PASNG adopts a loop mechanism where each loop contains K_m iterations. According to the results of repeat experiments, under the condition of K_m users, the number of rounds when the convergence is achieved is about K_m. Thus, the computational complexity of PASNG is $\mathcal{O}(K^2_m \times N_{pop} \times N_{iter})$. ■

B. Power Allocation of Matching Group

We investigate the power allocation problem where the NOMA group matches a D2D pair. We assume that the NG_m shares subchannel with the D_n. The power allocation problem of the matching group is given by

$$\mathbf{P4}: \min_{\{p_{NG^j_m}, j\in[1,K_m]\}, p_n} \omega\Big(\sum_{j=1}^{K_m} \frac{p_{NG^j_m} D_{NG^j_m}}{R_{NG^j_m}} + \frac{p_n D_n}{R_n}\Big) + (1-\omega)\Big(\sum_{j=1}^{K_m} \frac{D_{NG^j_m}}{R_{NG^j_m}} + \frac{D_n}{R_n} + \sum_{j=1}^{K_m} T^{exe}_{NG^j_m} + \frac{C_n}{F^{clo}_n}\Big), \tag{17a}$$

$$s.t. \quad E_{NG^j_m} \le E^{\max}_{NG^j_m}, \forall j \in [1, K_m], \tag{17b}$$

$$E_n \le E^{\max}_n, \tag{17c}$$

$$T_{NG^j_m} \le T^{\max}_{NG^j_m}, \forall j \in [1, K_m], \tag{17d}$$

$$T_n \le T^{\max}_n, \tag{17e}$$

$$p^{\min}_{NG^j_m} \le p_{NG^j_m} \le p^{\max}_{NG^j_m}, \forall j \in [1, K_m], \tag{17f}$$

$$p^{\min}_n \le p_n \le p^{\max}_n, \forall n \in [1, N], \tag{17g}$$

where $\sum_{j=1}^{K_m} T^{exe}_{NG^j_m}$ and $\frac{C_n}{F^{clo}_n}$ are constants. Since NG_m is a matching group, $\Gamma_{NG^j_m} = \frac{p_{NG^j_m} g_{NG^j_m}}{\sum_{l=j+1}^{K_m} p_{NG^l_m} g_{NG^l_m} + p_n g_{nB} + \sigma^2}$ and $\Gamma_n = \frac{p_n g_{nn}}{\sum_{j=1}^{K_m} p_{NG^j_m} g_{NG^j_m,n} + \sigma^2}$. This problem is a multivariable fractional summation problem. To address $\mathbf{P4}$, we fix the task delay of each user in NOMA group as x times of the optimal task delays obtained by the PASNG, where x is uniformly distributed in $(1, 2)$. Under the condition, the SINR of NG^j_m is given by

$$\gamma_{NG^j_m} = \frac{p_{NG^j_m} g_{NG^j_m}}{\sum_{l=j+1}^{K_m} p_{NG^l_m} g_{NG^l_m} + p_n g_{nB} + \sigma^2}, \tag{18}$$

where $\gamma_{NG^j_m} = 2^{\frac{D_{NG^j_m}}{BT^{up}_{NG^j_m}}} - 1$. According to (17), the power of each user in NOMA group is given by

$$p_{NG^{K_m}_m} = \frac{\gamma_{NG^{K_m}_m}(p_n g_{nB} + \sigma^2)}{g_{NG^{K_m}_m}}, \tag{19}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access

$$p_{NG_m^j} = \frac{\gamma_{NG_m^j}(\sum_{l=j+1}^{K_m} p_{NG_m^l} g_{NG_m^l} + p_n g_{nB} + \sigma^2)}{g_{NG_m^j}}, \forall j \in [1, K_m - 1]. \tag{20}$$

In order to completely represent the power of CUs in (19) and (20) by the power of the DTR_n, we introduce

$$S_{NG_m^j} = \sum_{l=j}^{K_m} p_{NG_m^l} g_{NG_m^l}, \forall j \in [1, K_m], \tag{21}$$

and according to (19) and (20), we can get the following formula

$$S_{NG_m^j} = G_{NG_m^j} S_{NG_m^{j+1}} + H_{NG_m^j}, \tag{22}$$

where $G_{NG_m^j} = \gamma_{NG_m^j} + 1$ and $H_{NG_m^j} = \gamma_{NG_m^j}(p_n g_{nB} + \sigma^2)$. We define $S_{NG_m^{K_m+1}} = 0$. Using the recursive method, (21) can be rewritten as

$$S_{NG_m^j} = \sum_{l=j}^{K_m} H_{NG_m^l} \prod_{o=j}^{l-1} G_{NG_m^o}, \tag{23}$$

and we define $\prod_{o=j}^{j-1} G_{NG_m^o} = 1$. Since $S_{NG_m^j} - S_{NG_m^{j+1}} = p_{NG_m^j} g_{NG_m^j}$, combining with (23), we can get the power of the CUs in NOMA group as

$$\begin{aligned} p_{NG_m^j} &= \frac{S_{NG_m^j} - S_{NG_m^{j+1}}}{g_{NG_m^j}} \\ &= \frac{\sum_{l=j}^{K_m} H_{NG_m^l} \prod_{o=j}^{l-1} G_{NG_m^o} - \sum_{l=j+1}^{K_m} H_{NG_m^l} \prod_{o=j+1}^{l-1} G_{NG_m^o}}{g_{NG_m^j}} \\ &= I_{NG_m^j} p_n + J_{NG_m^j}, \end{aligned} \tag{24}$$

where

$$\begin{cases} I_{NG_m^j} = \frac{\gamma_{NG_m^j} g_{NG_m^j} (1 + \sum_{l=j+1}^{K_m} \gamma_{NG_m^l} \prod_{o=j+1}^{l-1} (\gamma_{NG_m^o} + 1))}{g_{NG_m^j}} \\ J_{NG_m^j} = \frac{\gamma_{NG_m^j} \sigma^2 (1 + \sum_{l=j+1}^{K_m} \gamma_{NG_m^l} \prod_{o=j+1}^{l-1} (\gamma_{NG_m^o} + 1))}{g_{NG_m^j}} \end{cases}.$$

In summary, we convert the multivariate problem $\mathbf{P4}$ into a single variable problem by fixing the CUs' task delay and using the expression of DTR's power to replace the CUs' power. Therefore, $\mathbf{P4}$ can be rewritten as

$$\mathbf{P4.1}: \min_{p_n} \omega(\sum_{j=1}^{K_m} (I_{NG_m^j} p_n + J_{NG_m^j}) T^{exe}_{NG_m^j} + p_n \frac{D_n}{R_n}) + (1-\omega)\frac{D_n}{R_n} + (1-\omega)(T_{NG_m^j} + \frac{C_n}{F_n^{clo}}), \tag{25a}$$

$$s.t. \quad (I_{NG_m^j} p_n + H_{NG_m^j}) T^{up}_{NG_m^j} \le E^{\max}_{NG_m^j}, \forall j \in [1, K_m], \tag{25b}$$

$$E_n \le E_n^{\max}, \tag{25c}$$

$$T_n \le T_n^{\max}, \tag{25d}$$

$$p^{\min}_{NG_m^j} \le I_{NG_m^j} p_n + H_{NG_m^j} \le p^{\max}_{NG_m^j}, \tag{25e}$$

$$p_n^{\min} \le p_n \le p_n^{\max}, \tag{25f}$$

where $R_n = B\log_2(\frac{Qp_n+O}{Lp_n+O})$ and $L = \sum_{j=1}^{K_m} I_{NG_m^j} g_{NG_m^j,n}$. $O = \sum_{j=1}^{K_m} H_{NG_m^j} g_{NG_m^j,n} + \sigma^2$ and $Q = L + g_{nn}$. Since $(1-\omega)(T_{NG_m^j} + \frac{C_n}{F_n^{clo}})$ and $\omega \sum_{j=1}^{K_m} J_{NG_m^j} T^{exe}_{NG_m^j}$ are constants, they are not considered in the objective function of $\mathbf{P4.1}$. Moreover, we convert the constraints of $\mathbf{P4.1}$ into new expressions based on the DTR's power. Therefore, the simplified problem can be expressed as

$$\mathbf{P4.2}: \min_{p_n} \omega \sum_{j=1}^{K_m} I_{NG_m^j} T^{exe}_{NG_m^j} p_n + \frac{D_n(\omega p_n + (1-\omega))}{R_n}, \tag{26a}$$

$$s.t. \quad p_n \le \frac{E^{\max}_{NG_m^j} - H_{NG_m^j} T^{up}_{NG_m^j}}{T^{up}_{NG_m^j} I_{NG_m^j}}, \forall j \in [1, K_m], \tag{26b}$$

$$p_n \ge \frac{(2^{\frac{\varphi}{B}} - 1)O}{Q - 2^{\frac{\varphi}{B}} L}, \tag{26c}$$

$$\frac{p^{\min}_{NG_m^j} - H_{NG_m^j}}{I_{NG_m^j}} \le p_n \le \frac{p^{\max}_{NG_m^j} - H_{NG_m^j}}{I_{NG_m^j}}, \tag{26d}$$

$$p_n \ge \frac{(2^{\frac{\phi}{B}} - 1)O}{Q - 2^{\frac{\phi}{B}} L}, \tag{26e}$$

$$p_n^{\min} \le p_n \le p_n^{\max}, \tag{26f}$$

where $\varphi = \frac{p_n^{\max} D_n}{E_n^{\max}}$ and $\phi = \frac{D_n}{T_n^{\max}}$. Moreover, we define that $f(p_n) = R_n$, $z(p_n) = \frac{D_n(\omega p_n + (1-\omega))}{f(p_n)}$, and $h(p_n) = \omega f(p_n) - \omega p_n f'(p_n) - (1-\omega) f'(p_n)$. We assume that p_n^0 is the zero point which indicates $h(p_n^0) = 0$.

Proposition 5 The optimal interval for power allocation of DTR is $(0, p_n^0]$ under the condition that the task delays of all users in the NOMA group are fixed.

Proof: First, we get the first derivative of $f(p_n)$, the expression of $f'(p_n)$ is given by

$$\begin{aligned} f'(p_n) &= \frac{B}{\ln 2} \frac{O g_{nn}}{(Qp_n + O)(Lp_n + O)} \\ &= \frac{B}{\ln 2} \left(\frac{Q}{Qp_n + O} - \frac{L}{Lp_n + O} \right) \end{aligned}, \tag{27}$$

and then from the first derivative, we can get the second derivative

$$f''(p_n) = -\frac{B g_{nn}}{\ln 2} \frac{(2QLp_n + O(Q+L))}{(Qp_n + O)^2 (Lp_n + O)^2}. \tag{28}$$

Because Q, L, O, p_n are all positive, $(2QLp_n + O(Q+L))$ is always positive. Obviously, the second derivative of $f(p_n)$ is always negative. Moreover, the first derivative of $z(p_n)$ is given by

$$z'(p_n) = \frac{\omega f(p_n) - \omega p_n f'(p_n) - (1-\omega) f'(p_n)}{f^2(p_n)}. \tag{29}$$

Since $f^2(p_n)$ is always positive, we study the molecules of $z'(p_n)$. Then, first derivative of $h(p_n)$ is given by

$$\begin{aligned} h'(p_n) &= \omega f'(p_n) - \omega f'(p_n) - \omega p_n f''(p_n) - (1-\omega) f''(p_n) \\ &= -f''(p_n)(\omega p_n + (1-\omega)) \end{aligned}. \tag{30}$$

Since $f''(p_n)$ is always negative, $h(p_n)$ is a monotonically increasing function. Moreover, it's easy to know that

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access

$$\lim_{p_n\to 0} f(p_n) = \lim_{p_n\to 0} B\log_2(\frac{Qp_n+O}{Lp_n+O}) = 0, \tag{31a}$$

$$\lim_{p_n\to 0} f^{'}(p_n) = \frac{B}{\ln 2}\frac{Q-L}{O} = \frac{Bg_{nn}}{O\ln 2} > 0, \tag{31b}$$

$$\lim_{p_n\to +\infty} f(p_n) = \lim_{p_n\to +\infty} B\log_2(\frac{Qp_n+O}{Lp_n+O}) = B\log 2(\frac{Q}{L}) > 0, \tag{31c}$$

$$\lim_{p_n\to +\infty} f^{'}(p_n) = \lim_{p_n\to +\infty} \frac{B}{\ln 2}\left(\frac{Q}{Qp_n+O} - \frac{L}{Lp_n+O}\right) = 0, \tag{31d}$$

$$\lim_{p_n\to +\infty} p_n f^{'}(p_n) = \lim_{p_n\to +\infty} \frac{B}{\ln 2}\left(\frac{Qp_n}{Qp_n+O} - \frac{Lp_n}{Lp_n+O}\right) = 0. \tag{31e}$$

Therefore, $\lim_{p_n\to 0} h(p_n) = \lim_{p_n\to 0}[\omega f(p_n) - \omega p_n f^{'}(p_n) - (1-\omega)f^{'}(p_n)] < 0$ and $\lim_{p_n\to +\infty} h(p_n) = \lim_{p_n\to +\infty}[\omega f(p_n) - \omega p_n f^{'}(p_n) - (1-\omega)f^{'}(p_n)] = \omega B\log 2(\frac{Q}{L}) > 0$. Therefore, there must be a only $p_n^0 > 0$ that makes $h(p_n^0) = 0$. Thus, z is a function of p_n with a unique trough. Moreover, $z(\cdot)$ is monotonically decreasing in $(0, p_n^0)$, and is monotonically increasing in (p_n^0, ∞).

Besides, since $I_{NG_m^j}$ and $T_{NG_m^j}^{exe}$ are always positive, $\omega\sum_{j=1}^{K_m} I_{NG_m^j} T_{NG_m^j}^{exe} p_n$ is monotonically increasing in $(0,\infty)$. Hence, the objective function in $\mathbf{P4.2}$ is monotonically increasing in the interval $[p_n^0, +\infty)$. Since we need to find the minimum value of the cost, the optimal power of the DTR can only be distributed within $(0, p_n^0]$.

■

Based on the Proposition 5, jointly solving constraints (25b)-(25f), the feasible interval of DTR_n's power can be obtained. We assume that the feasible interval of DTR_n's power is $[p_n^{a,\min}, p_n^{a,\max}]$. In the interval, we can utilize intelligent algorithms to search the optimal power.

In summary, we propose a power allocation algorithm for the matching group (PAMG) to solve $\mathbf{P4.3}$. The pseudo-code for PAMG is summarized in Algorithm 3, where $\varepsilon_0 = 10^{-5}$ and $R_{NG_m^j} = B\log_2(1+\frac{p_{NG_m^j} g_{NG_m^j}}{\sum_{l=j+1}^{K_m} p_{NG_m^l} g_{NG_m^l} + \sigma^2})$. First, PASNG is used to obtain the optimal power of CUs in the NOMA group, and then the corresponding task delay is obtained from the optimal power. Then, we utilize the dichotomy [40] to obtain the zero point p_n^0 and solve the constraints (26b)-(26f) to obtain the feasible interval $[p_n^{a,nin}, p_n^{a,\max}]$. Next, we compare the $p_n^{a,\min}$ and p_n^0 to determine the search interval. Finally, we utilize the PSO to obtain the optimal power of the DTR_n and utilize (24) to calculate the optimal power of the CUs.

Algorithm 3 Power Allocation Algorithm for Matching Group (PAMG)

1: **Initialization:** Obtain the optimal users' power of the NOMA group NG_m by PASNG
2: Calculate the task uploading delay of each user in NG_m according to $T_{NG_m^j}^{up} = \frac{D_{NG_m^j}}{R_{NG_m^j}}$.
3: Utilize the dichotomy to find the approximate zero point p_n^0 and satisfy $\left|h(p_n^0)\right| < \varepsilon_0$.
4: Solve the constraints (26b)-(26f) to obtain the feasible interval of DTR's power $[p_n^{a,nin}, p_n^{a,\max}]$.
5: **if** $p_n^{a,\min} < p_n^0$
6: Search interval is $[p_n^{a,\min}, p_n^{a,\max}]$.
7: **else**
8: Search interval is $[p_n^{a,\min}, p_n^{a,\min}]$.
9: **end if**
10: Use the PSO to search the optimal DTR's power p_n^*.
11: Use formula (24) to calculate the optimal power of the CUs $\{p_{NG_m^j}^{m,*}, j\in[1,K_m]\}$.
12: **output:** $p_n^*, \{p_{NG_m^j}^{m,*}, j\in[1,K_m]\}$

V. Channel Allocation and Joint Algorithm

The D2D pairs are objects of the channel allocation, and each subchannel is occupied with a NOMA group. Hence, the channel allocation problem can be regarded as the matching problem between NOMA groups and D2D pairs. In the previous section, we have solved the power allocation problem for the single NOMA group and the matching group. In this section, we investigate the matching problem between NOMA groups and D2D pairs based on the results of the power allocation.

A. *Channel Allocation*

First, we consider a one-to-one matching scenario where the number of D2D pairs is less than or equal to the number of NOMA groups. To solve the channel allocation problem in the scenario, we propose a matching algorithm based on the Pareto improvement and swapping operations. To better understand the algorithm, we give some definitions.

Definition 1: In the one-to-one matching model, the matching Ω is a function of set $\mathcal{NG}\cup\mathcal{D}$. Some features of the function are shown below: 1) $|\Omega(D_n)| = 1, \forall n\in[1,N]$, 2) $\Omega(D_n) = NG_m$, 3) $|\Omega(NG_m)| \le 1, \forall m\in[1,M]$, 4) $\Omega(NG_m) = D_n$, if $|\Omega(NG_m)| = 1$.

It can be seen from definition 1 that a D2D pair must share a subchannel with a certain NOMA group, howover, a NOMA group can not match any D2D pair. Obviously, different matching schemes will result in different weight sums. To better analyze the different matching schemes, we define the utility function $U_n(\cdot)$ and $U_{NG_m}(\cdot)$ of D2D pairs and NOMA groups, respectively.

$$U_n(NG_m) = \omega\frac{p_n D_n}{R_n} + (1-\omega)(\frac{D_n}{R_n} + \frac{C_n}{F_n^{clo}}), \tag{32a}$$

$$U_{NG_m}(D_n) = \omega\sum_{j=1}^{K_m}\frac{p_{NG_m^j} D_{NG_m^j}}{R_{NG_m^j}} + (1-\omega)\sum_{j=1}^{K_m}(\frac{D_{NG_m^j}}{R_{NG_m^j}} + T_{NG_m^j}^{exe}), \tag{32b}$$

and due to the one-to-one matching state, the SINRs of NG_m^j and DTR_n are $\Gamma_{NG_m^j} = \frac{p_{NG_m^j} g_{NG_m^j}}{\sum_{l=j+1}^{K_m} p_{NG_m^l} g_{NG_m^l} + p_n g_{nB} + \sigma^2}$ and $\Gamma_n = \frac{p_n g_{nn}}{\sum_{j=1}^{K_m} p_{NG_m^j} g_{NG_{m,n}^j} + \sigma^2}$, respectively.

Next, we present the swapping operations between two D2D pairs which have been matched to different NOMA groups

respectively. The swapping operation means two D2D pairs swap with each other, which is given by

$$\Omega_{n\leftrightarrow n'} \triangleq \Omega(D_n) \leftrightarrow \Omega(D_{n'}). \tag{33}$$

Moreover, the two D2D pairs which can be swaped are called swapping pair. However, not all D2D pairs can form the swapping pair. The swapping pair is defined in the following:

Definition 2: $(D_n, D_{n'})$ *is a swapping pair if and only if*

1) $\forall i \in \{D_n, D_{n'}, \Omega(D_n), \Omega(D_{n'})\}, U_i(\Omega^{'}(i)) \leq U_i(\Omega(i))$,

2) $\exists i \in \{D_n, D_{n'}, \Omega(D_n), \Omega(D_{n'})\}, U_i(\Omega^{'}(i)) < U_i(\Omega(i))$.

In the definition 2, $\Omega^{'}$ is the matching after performing the swapping operation. As can be seen from the two points in the definition 2, if the utility functions of two D2D pairs satisfy the pareto improvement condition, they are a swapping pair. After the above description, we propose the one-to-one matching algorithm (OTOM) which is summarized in Algorithm 4. Specifically, we first randomly select N NOMA groups to participate in the matching. Then, we randomly match the N NOMA with N D2D pairs, and calculate the corresponding utility function value of the N matching groups by PAMG. Finally, we keep looking for swapping pairs and performing swapping operations until there is no any swapping pair.

Algorithm 4 One-to-one Matching Algorithm (OTOM)

1: **Initialization:**
2: Randomly selecte N NOMA groups and calculate their weight sum by PASNG.
3: Randomly and one-to-one match the N NOMA groups with N D2D pairs.
4: Calculate the utility function of the N matching group by PAMG.
5: **Swapping Operations:**
6: **Repeat:**
7: **for** $\forall D_n \in \mathcal{D}$ **do**
8: **for** $\forall D_{n'} \in \{\mathcal{D} \backslash D_n\}$ **do**
9: Calculate the utility function after the exchange operation by PAMG.
10: **if** $(D_{n'}, D_n)$ satisfy pareto improvement conditions **then**
11: Execute an swapping operation $\Omega_{n\leftrightarrow n'}$.
12: **break**
13: **end if**
14: **end for**
15: **end for**
16: **until** there is no swapping pair
17: **Output:** Optimized matching Ω^*

Proposition 6 The proposed one-to-one matching algorithm converges the locally or globally optimal matching.

Proof: We define that when the swapping operation is executed, the matching changes from Ω to $\Omega^{'}$. For Ω, supposing the weight sum of the system is $\Psi(\Omega)$. Because the swapping pair has to satisfy the pareto improvement condition, $\Psi(\Omega^{'}) < \Psi(\Omega)$. Hence, when each swapping operation is executed, the weight sum is reduced. Since the number of D2D pairs is limited, the total number of matching is limited. There must be a matching which makes the weight sum minimum. When the algorithm cannot find a new matching to reduce the weight sum, the algorithm converges to the locally or globally optimal matching. ■

Proposition 7 The computational complexity of proposed one-to-one matching algorithm is $\mathcal{O}(K_m^2 \times N_{pop} \times N_{iter} + \frac{\Psi_{\Omega^0\rightarrow\Omega^*}}{\Delta_{average}})$.

Proof: In the initialization of OTOM, we utilize PASNG whose computational complexity is $\mathcal{O}(K_m^2 \times N_{pop} \times N_{iter})$ to calculate the weight sum of the N NOMA groups. Then, we assume the initial matching is Ω^0 and the gap of weight sum between the initial matching and the optimal matching is $\Psi_{\Omega^0\rightarrow\Omega^*}$. Moreover, we assume that average variation of weight sum in the swapping operation is $\Delta_{average}$. Hence, the average number of swapping operations is $\frac{\Psi_{\Omega^0\rightarrow\Omega^*}}{\Delta_{average}}$. Therefore, the computational complexity of proposed one-to-one matching algorithm is $\mathcal{O}(K_m^2 \times N_{pop} \times N_{iter} + \frac{\Psi_{\Omega^0\rightarrow\Omega^*}}{\Delta_{average}})$. ■

When the number of D2D users is greater than the number of NOMA groups, the one-to-one matching will no longer apply because there will always be a surplus of D2D pairs which are idle, which greatly increases the task delay. Thus, we consider the many-to-one matching. Moreover, due to complexity, we only consider the situation where two D2D pairs match a NOMA group, and assume that the number of D2D pairs does not exceed twice the number of NOMA groups.

Next, based on the OTOM, we propose the many-to-one matching algorithm (MTOM). Specifically, if the number of D2D pairs is less than or equal to the number of NOMA groups, we randomly select $N-1$ D2D pairs and NOMA groups. Next, we utilize OTOM to obtain the optimal matching for the selected NOMA groups and D2D pairs. For each new group consisting of a matching group and a remaining D2D pair, we fix the power of the matching group to the results of optimization by PAMG, and then use the PSO algorithm to optimize the power of the DTR in the remaining D2D pair. Based on this method, we calculate the weight sum of the remaining D2D pairs with each matching group, and select the matching scheme with the minimum weight sum for each remaining D2D pair. Moreover, when the number of D2D pairs is greater than the number of NOMA groups, M D2D pairs are randomly selected to match all NOMA groups, and their optimal matching is obtained by OTOM. For the remaining D2D pairs, the weight sum of each D2D pair with each matching group is calculated, and then we select the matching scheme with the minimum weight sum. The convergence and complexity analysis of many-to-one matching is similar to OTOM. Therefore, analysis is omitted.

B. Joint Computing Resource, Power and Channel Allocation Algorithm

Based on the proposed algorithms for computing resource, power and channel allocations, we combine the three algorithms to solve the original problem. We propose a scheduling-based joint computing resource, power and channel allocations algorithm (S-JCRPCA) whose flow has been summarized in Fig. 2.

Specifically, we first utilize CRA to solve the problem of computing resource allocation. Then, we propose a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access

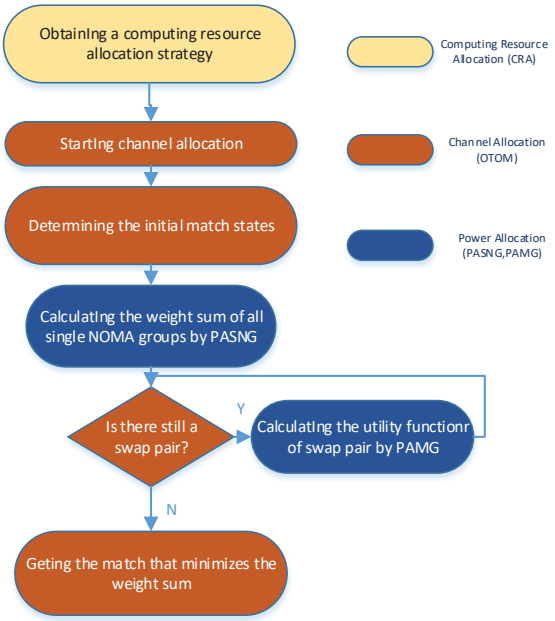


Fig. 2. Scheduling-based joint computing resource, power and channel allocation algorithm (S-JCRPCA) flow

scheduling-based combination of power and channel allocation algorithms. In this combination, the power allocation algorithm is scheduled by the channel allocation algorithm. In the beginning of the channel allocation algorithm, we determine the initial matching states. Next, for the single NOMA groups, since their matching states no longer change, we calculate their weight sum by scheduling the PASNG. For the matching groups, since the matching groups will converge to the locally or globally optimal solution by the swapping operations, we schedule the PAMG to calculate the utility function of swapping pairs in swapping operations. Moreover, it is worth mentioning that the power allocation algorithm will be repeatedly scheduled until there is no any swapping pair. Such that, the channel allocation algorithm ends. In summary, the S-JCRPCA adopts a scheduling-based method where Pareto improvement and swapping operations are introduced to ensure local or global optimality of the solution.

VI. Numerical Results

In this section, we investigate the performance of D2D-assisted and NOMA-based MEC system. We consider a cellular uplink communication system. The distance between DTR and DTA in each D2D pair is uniformly distributed between $[1, 3]$ (in meters). Rayleigh fading model is considered in the system, where the channel gains are exponentially distributed with unit-mean. The calculation rate of each DTA is uniformly distributed in $[1.8 - 2.4]$ (in gigacycles per second). The computing rate and total number of the computing cell are 10 gigacycles per second and 50, respectively. The size of each task is randomly generated between 0.1 Mbits and 0.5 Mbits. The rest parameter value settings are summarized in Table I.

TABLE I
Simulation Parameters

Cellular radius	$250m$
Maximum transmission power	$24dBm$
Minimum transmission power	$-20dBm$
Maximum tolerable energy consumption	$100mJ$
Maximum tolerance task delay	$500ms$
Pass-loss exponent	4
Noise power	$-174dBm$
CPU cycles per bit	$1000cps$
The populations of PSO	20
Number of iterations of PSO	50

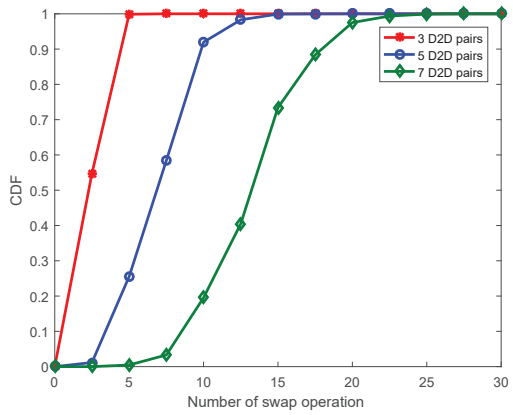


Fig. 3. CDF of the number of swapping operations, with M=10, U=20, ω=0.6

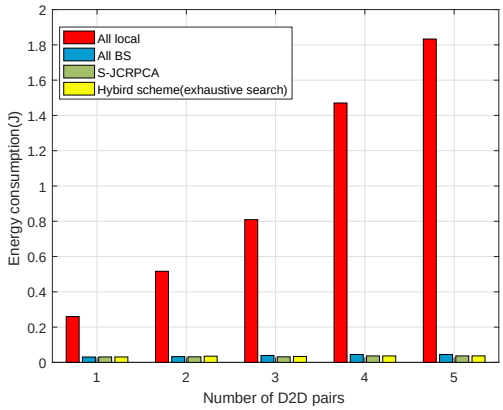
A. *Convergence of the S-JCRPCA Algorithm*

In Fig. 3, we plot the cumulative distribution function (CDF) of the number of swapping operations when the number of D2D pair is 3, 5, 7, respectively. The curves are obtained by simulating 10000 independent trials. It is worth mentioning that since the proposed S-JCRPCA is based on the scheduling manner, its convergence is equivalent to the convergence of the channel allocation algorithm. As shown in the figure, the CDF can achieve convergence with few number of swapping operations in different number of D2D pairs, which indicates the convergent ability of the channel allocation algorithm, and thus proves the convergence of the S-JCRPCA algorithm. Besides, as the number of D2D pairs increases, the number of swapping operations when the CDF converges to 1 continuously increases. This is because when the number of D2D pairs increases, there are more swapping pairs, and more swapping operations will be performed.

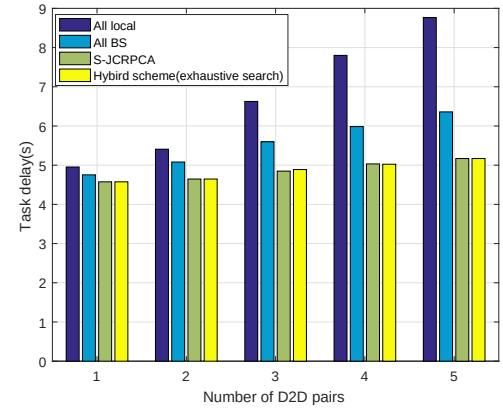
B. *Performance Evaluation for Different Offloading Schemes*

Fig.4 plots the system performance of different offloading schemes. As shown in Fig. 4, we study four kinds of task execution schemes for DTRs. They are 1)All local: the tasks of all DTRs are executed locally, 2)All BS: all DTRs offload tasks to the edge server, 3)S-JCRPCA: all DTR offload tasks to the corresponding DTA, and each NOMA group shares the subchannel with at most one D2D pair, 4)Hybrid scheme: each

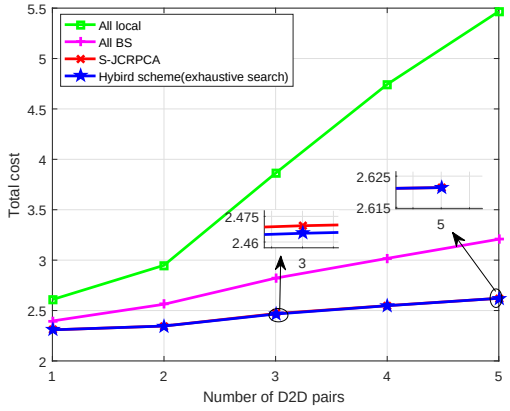
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access



(a) Energy consumption



(b) Task delay



(c) Total cost

Fig. 4. Performance comparison of different offloading schemes, with M = 5, U=25, ω=0.5

DTR can choose one of the above three schemes, and the scheme minimizing the total cost will be adopted.

It can be seen from Fig. 4(a) that the energy consumption of "All local" is much higher than that of the other three schemes, which shows that local computing is more energy intensive than computing offloading. Moreover, the energy consumption of the "All BS" is slightly higher than that of the "S-JCRPCA", which shows that the D2D-assisted computing offloading consumes less energy than the general computing offloading where all tasks offload to the edge server.

Fig. 4(b) shows that as the number of D2D pairs increases, the delays of the "All local" and the "All BS" increase rapidly, while the delay of the "S-JCRPCA" increases slowly. This is because D2D-assisted computing offloading can control delay to a lower level compared to the local computing and the general computing offloading.

Moreover, it can be seen from Fig. 4(c) that the curves of the "S-JCRPCA" and the "Hybrid scheme" almost coincide. From the enlarged parts, it can be seen that they still have slight misalignment which is caused by the random selection when the one-to-one matching algorithm is executed. In summary, the minimum total cost can be obtained by offloading tasks of all DTRs to the corresponding DTA.

C. Performance Evaluation for Different Algorithm based on Different Access Manners

Fig. 5(a) shows that in any same matching scheme, the energy consumption of OMA-based MEC is significantly greater than that of NOMA-based MEC. This is because NOMA can enable multiple users to share greater bandwidth than that of OMA. Although interference will be introduced, the gain of greater bandwidth is significantly higher than the negative impact of interference. Moreover, in the same access mode, we find that when the number of D2D pairs increases, the energy consumption of the S-JCRPCA(one-to-one matching algorithm) is generally higher than that of the S-JCRPCA(many-to-one matching algorithm), which is because the interference of the single NOMA group is less than that of the matching NOMA group.

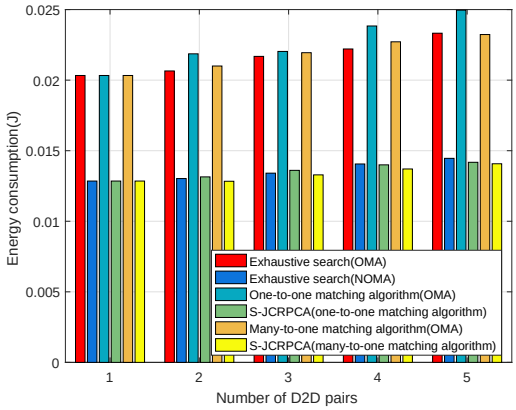
Fig. 5(b) shows that with the increase of D2D pairs' number, although the task delays of different algorithms in the same access mode basically keep the same value, the delay of S-JCRPCA(one-to-one matching algorithm) is less than that of S-JCRPCA(many-to-one matching algorithm). This shows that in the aspect of task delay, S-JCRPCA(many-to-one matching algorithm) will bring greater negative effects due to the interference between D2D pairs. In addition, the task delay of NOMA-based MEC is always lower than that of OMA-based MEC. Moreover, Fig. 5(c) indicates that the total cost of NOMA-based MEC is always less than that of OMA-based MEC, which shows that NOMA is superior to OMA in energy consumption and delay. Moreover, the curve of S-JCRPCA(one-to-one matching algorithm) is close to that of exhaustive search, which indicates that we can obtain the close-to-optimum solution by the S-JCRPCA.

When the number of D2D pairs is larger than that of NOMA groups, the one-to-one matching algorithm will no longer apply. Therefore, we investigate the many-to-one matching algorithm. To evaluate the performance of the proposed many-to-one matching algorithm, we plot the total cost in different many-to-one matching algorithms in Fig. 6. We find that even in the case where the number of D2D pairs is larger than that of NOMA groups, the total cost of NOMA-based MEC is always better than that of OMA-based MEC.

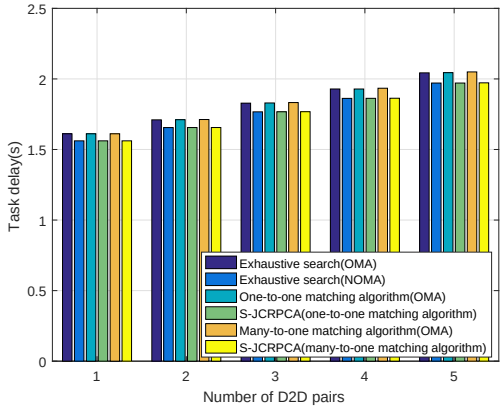
D. Performance Evaluation for Different Weight Factors

We plot the total cost in different weight factors in Fig. 7 to evaluate the performance of different weight factors. Fig.7

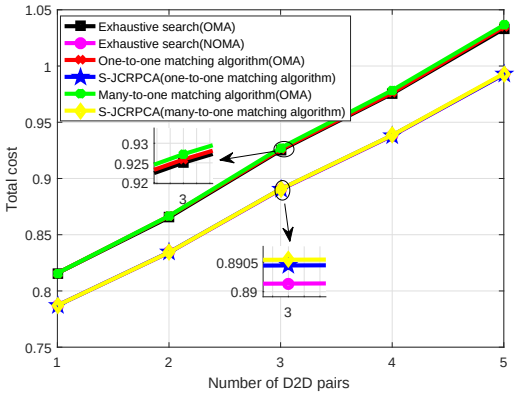
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2890559, IEEE Access



(a) Energy consumption



(b) Task delay



(c) Total cost

Fig. 5. Performance comparison of NOMA-based and OMA-based algorithms, with M = 5, U=15, ω=0.5

shows the total cost of different weight factors and the number of CUs. First, under the same weight factor, the total cost increases as the number of CUs increases. In addition, we find that when the number of CUs increases, the difference between NOMA and OMA under the same weight factor would be greater. This shows that the greater the number of CUs, the more obvious the advantage of NOMA relative to OMA. In the case of the same number of CUs, if the weight factor is smaller, the total cost will be larger. This is because when the weight factor is smaller, the proportion of task delay increases

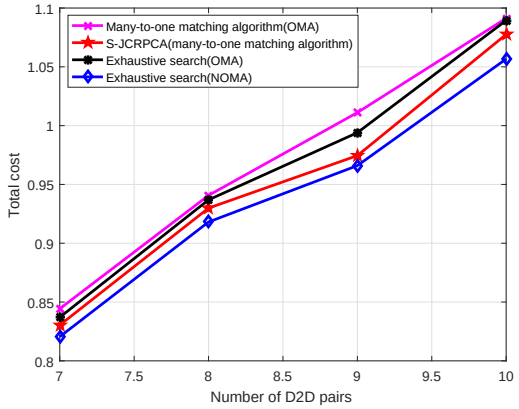


Fig. 6. Performance comparison of many-to-one matching algorithms, with M = 5, U=10, ω=0.5

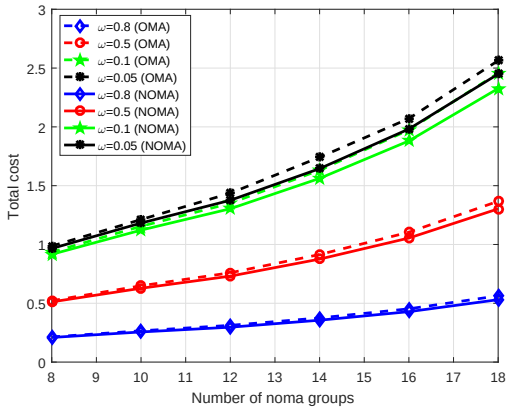


Fig. 7. Total cost versus number of noma groups, with N=5, M=5

and the absolute value of task delay is larger, so the total cost will increase.

VII. Conclusions

In this paper, we have studied the computing resource, power and channel allocations for a D2D-assisted and NOMA-based MEC system. Firstly, we have proposed a computing resource allocation algorithm to minimize the task execution delay. Secondly, we have utilized the PASNG to optimize the power of the single NOMA group. Then, we have derived the interval of optimal power allocation for DTRs and optimized the power of all users in the matching group. Next, we have proposed a one-to-one matching algorithm and extended it to a many-to-one situation. Finally, we have proposed a scheduling-based joint algorithm to solve the original optimization problem. Simulation results showed that the proposed S-JCRPCA can effectively reduce the total cost. Meanwhile, in terms of the total cost, the D2D-assisted computing offloading outperforms other computing modes, and in terms of the weight sum of the system, NOMA-based MEC system outperforms OMA-based MEC system. In the future work, for the D2D-assisted and NOMA-based MEC system, distributed power and channel allocation algorithms will be studied to reduce the information interaction between base stations and users.

REFERENCES

[1] Y. Xu and S. Mao, "A survey of mobile cloud computing for rich media applications," *IEEE Wireless Commun.*, vol. 20, no. 3, pp. 46-53, Jun. 2013.

[2] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, and R. Buyya, "Cloud-based augmentation for mobile devices: Motivation, taxonomies, and open challenges," *IEEE Commun. Surveys Tut.*, vol. 16, no. 1, pp. 337-368, 1st quarter 2014.

[3] Y. Wu, L. Qian, J. Zheng, H. Zhou, and X. Shen, "Green-oriented traffic offloading through dual-connectivity in future heterogeneous small-cell networks," *IEEE Commun. Mag.*, vol. 56, no. 5, pp. 140-147, May. 2018.

[4] J. Zheng, Y. Cai, Y. Wu, and X. Shen, "Dynamic computation offloading for mobile cloud computing: A stochastic game-theoretic approach," *IEEE Trans. Mobile Comput.*, to be published.

[5] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tut.*, vol. 19, no. 4, pp. 2322-2358, 4th quarter 2017.

[6] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tut.*, vol. 19, no. 3, pp. 1628-1656, 3rd quarter 2017.

[7] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757-6779, Mar. 2017.

[8] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks," *IEEE Signal Proc. Mag.*, vol. 31, no. 6, pp. 45-55, Nov. 2014.

[9] S. M. R. Islam, N. Avazov, O. A. Dobre, and K. Kwak, "Power-domain non-orthogonal multiple access (NOMA) in 5G systems: Potentials and challenges," *IEEE Commun. Surveys Tut.*, vol. 19, no. 2, pp. 721-742, 2nd quarter 2017.

[10] Y. Wu, K. Ni, C. Zhang, L. Qian, and D. H. K. Tsang, "NOMA assisted multi-access mobile edge computing: A joint optimization of computation offloading and time allocation," *IEEE Trans. Veh. Technol.*, to be published.

[11] L. Dai, B. Wang, Y. Yuan, S. Han, C. I, and Z. Wang, "Non-orthogonal multiple access for 5G: Solutions, challenges, opportunities, and future research trends," *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 74-81, Sep. 2015.

[12] Y. Wu, L. Qian, H. Mao, X. Yang, and X. Shen, "Optimal power allocation and scheduling for non-orthogonal multiple access relay-assisted networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 11, pp. 2591-2606, Nov. 2018.

[13] Y. Liu, Z. Ding, M. Elkashlan, and H. V. Poor, "Cooperative non-orthogonal multiple access with simultaneous wireless information and power transfer," *IEEE J. Sel. Area. Commun.*, vol. 34, no. 4, pp. 938-953, Apr. 2016.

[14] A. Al-Shuwaili and O. Simeone, "Energy-efficient resource allocation for mobile edge computing-based augmented reality applications," *IEEE Wireless Commun. Letters*, vol. 6, no. 3, pp. 398-401, Apr. 2017.

[15] W. Fan, Y. Liu, B. Tang, F. Wu, and Z. Wang, "Computation offloading based on cooperations of mobile edge computing-enabled base stations," *IEEE Access*, vol. 6, pp. 22622-22633, Dec. 2017.

[16] L. Chen, S. Zhou, and J. Xu, "Computation peer offloading for energy-constrained mobile edge computing in small-cell networks," *IEEE/ACM Trans. Net.*, vol. 26, no. 4, pp. 1619 -1632, Jun. 2018.

[17] Z. Ning, P. Dong, X. Kong, and F. Xia, "A cooperative partial computation offloading scheme for mobile edge computing enabled Internet of Things," *IEEE Internet Things J.*, to be published.

[18] R. Beraldi, A. Mtibaa, and H. Alnuweiri, "Cooperative load balancing scheme for edge computing resources," in *proc. IEEE FMEC*, Jun. 2017, pp. 94-100.

[19] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 74-80, Feb. 2014.

[20] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Commun. Surveys Tut.*, vol. 16, no. 4, pp. 1801-1819, 4th quarter 2014.

[21] G. Fodor *et al.*, "Design aspects of network assisted device-to-device communications," *IEEE Commun. Mag.*, vol. 50, no. 3, pp. 170-177, Mar. 2012.

[22] X. Lin, J. G. Andrews, and A. Ghosh, "Spectrum sharing for device-to-device communication in cellular networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 12, pp. 6727-6740, Dec. 2014.

[23] C. Ma, W. Wu, Y. Cui, and X. Wang, "On the performance of successive interference cancellation in D2D-enabled cellular networks," in *Proc. IEEE INFOCOM*, Apr. 2015, pp. 37-45.

[24] Y. Ai, L. Wang, B. Jiao, and K. Chen, "Exploiting NOMA into socially enabled computation offloading," in *Proc. IEEE WCSP*, Oct. 2017, pp. 1-6.

[25] F. Wang, J. Xu, and Z. Ding, "Optimized multiuser computation offloading with multi-antenna NOMA," in *Proc. IEEE GC Wkshps*, Dec. 2017, pp. 1-7.

[26] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAV-mounted cloudlet: Optimization of bit allocation and path planning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2049-2063, May. 2017.

[27] A. Kiani and N. Ansari, "Edge computing aware NOMA for 5G networks," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1299-1306, Apr. 2018.

[28] X. Chen, L. Pu, L. Gao, W. Wu, and D. Wu, "Exploiting massive D2D collaboration for energy-efficient mobile edge computing," *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 64-71, Aug. 2017.

[29] J. Zhao, Y. Liu, K. K. Chai, Y. Chen, and M. Elkashlan, "Joint subchannel and power allocation for NOMA enhanced D2D communications," *IEEE Trans. Commun.*, vol. 65, no. 11, pp. 5081-5094, Nov. 2017.

[30] L. Pei *et al.*, "Energy-efficient D2D communications underlaying NOMA-based networks with energy harvesting," *IEEE Commun. Letters*, vol. 22, no. 5, pp. 914-917, May. 2018.

[31] H. Zheng, S. Hou, H. Li, Z. Song, and Y. Hao, "Power allocation and user clustering for uplink MC-NOMA in D2D underlaid cellular networks," *IEEE Wireless Commun. Letters*, to be published.

[32] L. Lei, D. Yuan, C. K. Ho, and S. Sun, "Power and channel allocation for non-orthogonal multiple access in 5G systems: Tractability and computation," *IEEE Trans. Wireless Commun.*, vol. 15, no. 12, pp. 8580-8594, Dec. 2016.

[33] Y. Pan, C. Pan, Z. Yang, and M. Chen, "Resource allocation for D2D communications underlaying a NOMA-based cellular network," *IEEE Wireless Commun. Letters*, vol. 7, no. 1, pp. 130-133, Feb. 2018.

[34] M. S Ali, H. Tabassum, and E. Hossain, "Dynamic user clustering and power allocation for uplink and downlink non-orthogonal multiple access (NOMA) systems," *IEEE Access*, vol. 4, no. 99, pp. 6325-6343, Aug. 2016.

[35] K. Zhang *et al.*, "Energy-efficient offloading for mobile edge computing in 5G heterogeneous networks," *IEEE Access*, vol. 4, pp. 5896-5907, Aug. 2016.

[36] P. Zhao, H. Tian, C. Qin, and G. Nie, "Energy-saving offloading by jointly allocating radio and computational resources for mobile edge computing," *IEEE Access*, vol. 5, pp. 11255-11268, Jun. 2017.

[37] J. Zhang *et al.*, "Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks," *IEEE Internet Things J.*, vol. 5, pp. 2633-2645, Aug. 2018.

[38] J. Zhang, W. Xia, F. Yan, and L. Shen, "Joint computation offloading and resource allocation optimization in heterogeneous networks with mobile edge computing," *IEEE Access*, vol. 6, pp. 19324-19337, Mar. 2018.

[39] F. Guo, H. Zhang, H. Ji, X. Li, and V. C. M. Leung, "An efficient computation offloading management scheme in the densely deployed small cell networks with mobile edge computing," *IEEE/ACM Trans. Net.*, to be published.

[40] S. Kuzuoka and S. Watanabe, "A dichotomy of functions in distributed coding: An information spectral approach," *IEEE Trans. Inform. Theory*, vol. 61, pp. 5028-5041, Jul. 2015.