

RESEARCH

Open Access



Computation efficiency optimization in UAV-enabled mobile edge computing system with multi-carrier non-orthogonal multiple access

Fangcheng Xu¹, Xiangbin Yu^{1,2*}, Jiali Cai¹ and Guangying Wang¹

*Correspondence:

yxbxwy@gmail.com

¹College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, 29 Jiangjun Ave., 210016 Nanjing, China

²Key Laboratory of Wireless Sensor Network and Communication, Shanghai Institute of Microsystem and Information Technology of Chinese Academy of Sciences, 865 Changning Road, 200050 Shanghai, China

Abstract

In this paper, we study the issue of fair resource optimization for an unmanned aerial vehicle (UAV)-enabled mobile edge computing (MEC) system with multi-carrier non-orthogonal multiple access (MC-NOMA). A computation efficiency (CE) optimization problem based on the max-min fairness principle under the partial offloading mode is formulated by optimizing the subchannel assignment, the local CPU frequency, and the transmission power jointly. The formulated problem belongs to the non-convex mixed integer nonlinear programming (MINLP), that is NP-hard to find the global optimal solution. Therefore, we design a polynomial-time algorithm based on the big-M reformulation, the penalized sequential convex programming, and the general Dinkelbach's method, which can choose an arbitrary point as the initial point and eventually converge to a feasible suboptimal solution. The proposed algorithm framework can be also applied to computation offloading only mode. Additionally, we derive the closed-form optimal solution under the local computing only mode. Simulation results validate the convergence performance of the proposed algorithm. Moreover, the proposed partial offloading mode with the CE maximization scheme outperforms that with the computation bits (CB) maximization scheme with respect to CE, and it can achieve higher CE than the benchmark computing modes. Furthermore, the proposed MC-NOMA scheme can attain better CE performance than the conventional OFDMA scheme.

Keywords: Unmanned aerial vehicle, Mobile edge computing, Non-orthogonal multiple access, Fair resource allocation design, Computation efficiency

1 Introduction

In the past decade, the rapid growth of mobile communication business has promoted the great progress of wireless communication and network technologies, which gives birth to mobile cloud computing (MCC), i.e., allowing computing tasks to run remotely at cloud data centers [1]. However, the user equipment (UE) is usually far away from the cloud data center, which may result in a long time of data exchange between them. As a consequence,

MCC is difficult to meet the computing service requirements of various real-time mobile applications [2]. Recently, Internet of Things (IoT) has been applied to many aspects, which can connect things via networks [3]. In contrast to the conventional MCC, mobile edge computing (MEC) has potential wide application ranges for IoT as a newly emerging technology [4]. The key idea of MEC is to decentralize computing servers at the wireless network edge closing to UEs so as to improve the computation capability of UEs and reduce the corresponding task delay and consumed energy greatly.

On the one hand, non-orthogonal multiple access (NOMA) is considered as a promising multiple access scheme in the future wireless communication systems [5]. The fundamental concept of NOMA is to serve more than one UE in one orthogonal resource block (ORB) by using superposition coding and multi-user detection technologies [6]. For instance, power-domain NOMA assigns different power levels to the UEs at the transmitter and exploiting successive interference cancellation (SIC) technology to decode overlapping signals at the receiver [7]. In [8], a NOMA-based resource allocation algorithm was designed for a cluster-based cognitive industrial IoT. This is very different from orthogonal multiple access (OMA) schemes, e.g., time division multiple access (TDMA), orthogonal frequency division multiple access (OFDMA), which service one UE at most in a time-domain ORB and a frequency-domain ORB, respectively.

However, the system complexity and decoding delay will become high if all users are multiplexed in one ORB. To this end, multi-carrier NOMA (MC-NOMA), i.e., the combination of NOMA and OFDMA, can be employed to meet the needs of massive connectivity and improve the transmission efficiency in MEC systems [9–12], where all users are divided into several NOMA groups and one NOMA group can occupy one ORB. In [9], a MEC-NOMA technique was proposed to minimize the consumed energy of MEC users, where the users can simultaneously upload their tasks via the uplink NOMA protocol. In [10], the total computation overhead minimization problem was investigated by coalition formation game theory in a MC-NOMA based MEC system. Moreover, [11] and [12] provided their separate theories and simulation results to the energy minimization in MC-NOMA based MEC systems, which show the application of MC-NOMA in MEC can improve the system performance greatly when compared to the OMA counterpart.

On the other hand, unmanned aerial vehicle (UAV) has attracted a great quantity of research interests in the wireless communication. The benefit of UAV as the aerial base station (BS) is great when compared with the traditional ground BS, such as flexible movement, high probability of line-of-sight (LoS) channel, and low cost and easy deployment. Therefore, MEC servers can be carried to UAV for the purpose of providing flexible computing services [13–18]. In [13], the Lyapunov optimization framework was invoked for minimizing the average weighted consumed energy of a UAV-MEC system. In [14], the resource optimization for maximizing computation rate was firstly studied in a wireless powered UAV-MEC system. In [15], user association and path planning were studied in a UAV-MEC system with energy restriction. In [16], a double deep Q-network based algorithm was developed to maximize the system reward of a UAV-MEC network. Moreover, [17] studied the sum-power optimization for a MEC system with several UAVs, which was addressed by solving the decomposed subproblems iteratively. Besides, [18] minimized the consumed energy of a UAV-MEC system, which can enable fairness among all the users.

It can be seen that the delay minimization, the energy consumption minimization, and the computation bits (CB) maximization are the main focus of resource allocation in MEC systems. However, with the rapid growth of communication business, the communication industry is becoming a major energy consumer. Against this background, it is critically important to consider green communication so as to improve the efficiency of resource utilization per joule. Particularly, it is suggested that future MEC pursue the maximization of computation efficiency (CE), i.e., the ratio of the total CB to the associated energy consumption [19–23]. In [19], the max-min CE optimization problem was studied in a MEC system powered by wireless energy. In contrast to [19, 20] maximized the weighted sum of CE in a MEC system. While TDMA is considered in both [19] and [20, 21] and [22] extended the CE maximization frameworks to MEC networks under NOMA and OFDMA, respectively. Furthermore, [23] aimed to maximize the CE of a multi-UAV based MEC system.

Motivated by the above discussions, we study the CE optimization problem in a UAV-MEC system with MC-NOMA, which is oriented towards computation-efficient massive connections and flexible deployment in the future MEC network.

The rest of this paper is organized as follows. The system model and problem formulation are presented in Section 2. The proposed CE optimization algorithm was detailed in Section 3. Two benchmark computing modes are discussed in Section 4. The relevant simulation results are provided in Section 5. Finally, the conclusion of this paper is made in Section 6.

2 Method

In this paper, we focus on a computation-efficient UAV-enabled and MC-NOMA based MEC system, where a UAV is fitted with a MEC server to supply computing services for the ground UEs via the LoS connections. Besides, MC-NOMA is employed for the uplink transmissions between all the UEs and the UAV, in which all the UEs are divided into several user groups without repetition, and the UEs in the same group can adopt uplink NOMA to upload their tasks simultaneously to the UAV-MEC server in one ORB.

On this basis, we formulate the CE optimization problem based on the max-min fairness principle under the partial offloading mode by optimizing the subchannel assignment, the UE's CPU frequency, and the transmission power jointly. Considering that the formulated problem belongs to the non-convex mixed-integer nonlinear programming (MINLP), we design a two-layer iterative algorithm based on the big-M reformulation, the penalized sequential convex programming (SCP), and the general Dinkelbach's method to enjoy a polynomial-time complexity. The proposed algorithm can start from an arbitrary point and eventually converge to a feasible suboptimal solution. Moreover, the other two typical computing mode are discussed as benchmarks, including the local computing only mode and the computation offloading only mode. Specially, the closed-form best optimal solution is derived under the local computing only mode, while a suboptimal solution is obtained using the above algorithm under the computation offloading only mode.

Finally, simulation results confirm that the proposed algorithm achieves good convergence performance and is robust to random initial points. In addition, the proposed CE maximization framework has obvious computation-efficient advantages over the CB maximization framework under the partial offloading mode. Furthermore, the CE performance of the partial offloading mode is superior to the benchmarks computing modes.

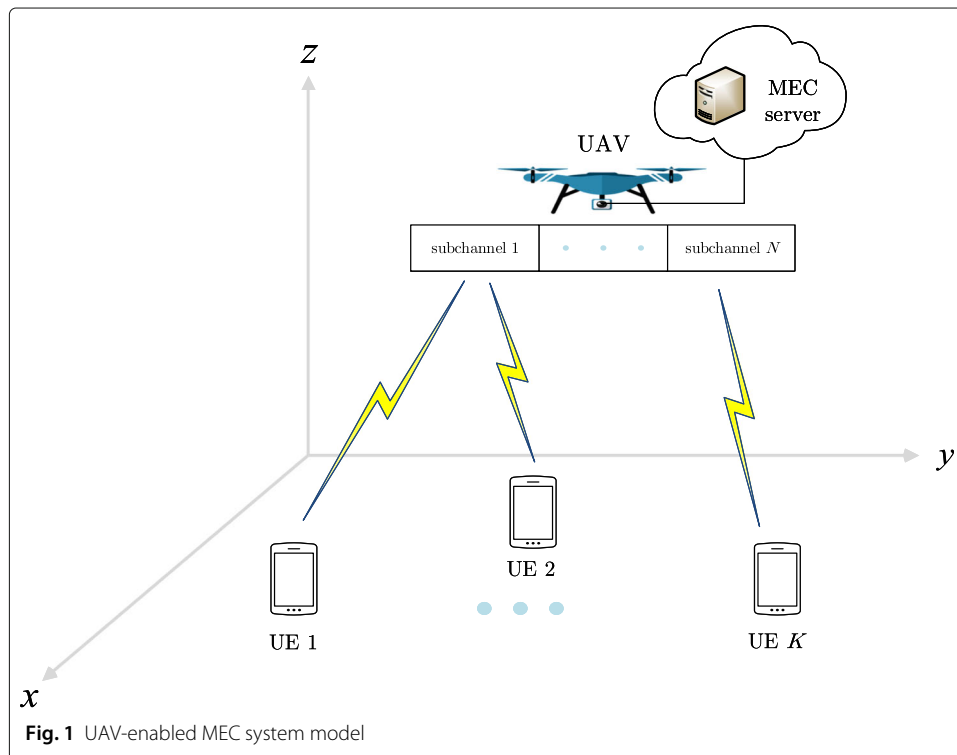
Moreover, the MC-NOMA scheme exhibits higher CE than the OFDMA scheme in the proposed UAV-MEC system.

3 System model and problem formulation

As illustrated in Fig. 1, we focus on a UAV-MEC system where one rotary-wing UAV with an embedded MEC server provides computing services for K ground UEs denoted by the set $k \in \mathcal{K} = \{1, 2, \dots, K\}$, in which the UAV and all K UEs are each equipped with one single-antenna. We construct a three-dimension (3D) Cartesian coordinate model, where the UAV and the k th UE are located at $\mathbf{q}_u = [X, Y, H]$ and $\mathbf{q}_k = [x_k, y_k, 0]$, $\forall k \in \mathcal{K}$, respectively. As a result, the distance between the UAV and the k th UE is given by $d_k = \sqrt{(X - x_k)^2 + (Y - y_k)^2 + H^2}$, $\forall k \in \mathcal{K}$.

3.1 Communication model

In the proposed UAV-MEC system, the overall system bandwidth W is partitioned into N orthogonal subchannels denoted by the set $n \in \mathcal{N} = \{1, 2, \dots, N\}$ satisfying $\sum_{n \in \mathcal{N}} W_n = W$, where W_n is the bandwidth of the n th subchannel. Besides, each subchannel is the frequency-flat block fading channel, i.e., the subchannel remains unchanged during one time block with duration T ($T > 0$). We assume the UAV knows the perfect channel state information (CSI) of all subchannels as well as the computation information of all the UEs. Moreover, it is assumed that all subchannels are each dominated by LoS links since the altitude of the UAV is usually much higher than that of the UEs. Hence, by adopting the commonly used free-space path loss model [13–18], the channel power gain on the n th subchannel between the k th UE and the UAV can be modeled as



$$g_{k,n} = \frac{\mu_0}{d_k^2} = \frac{\mu_0}{(X - x_k)^2 + (Y - y_k)^2 + H^2}, \tag{1}$$

where μ_0 represents the channel power gain at the reference distance $d_0 = 1$ m.

Particularly, one time block with duration T is composed of the following three stages. In the first stage, MC-NOMA is invoked for the uplink transmissions between the UEs and the UAV, i.e., each orthogonal subchannel can accommodate multiple users by employing NOMA. Specially, all the UEs are divided into different groups to upload their tasks simultaneously to the UAV-MEC server, where different groups access different subchannels and the UEs in the same group occupy one subchannel via NOMA. In the second stage, the UAV exploits SIC technique to decode the uploaded tasks from the UEs and the UAV-MEC server computes the decoded tasks. Finally, the UAV transmits the task results computed by the UAV-MEC sever to all the UEs in the third stage. Besides, each UE can perform its local computing throughout the entire time block duration T . Similar to [19–23], we only consider the first stage due to the strong computational capability of the UAV-MEC sever as well as the small data size of the computed results.

3.2 Local computing

For the local computing at the UEs, the total CB L_k^{loc} and total energy consumption E_k^{loc} of the k th UE during the entire time block T can be, respectively, expressed as [19–23]

$$L_k^{\text{loc}} = \frac{Tf_k}{C_k}, \tag{2a}$$

$$E_k^{\text{loc}} = \gamma_k f_k^3 T, \tag{2b}$$

where C_k denotes the CPU cycles of computing 1-bit of input task of the k th UE, f_k (cycles/second) is the CPU frequency at the k th UE, and γ_k is the CPU capacitance coefficient of the k th UE.

3.3 Computation offloading

To enable parallel execution, the proposed UAV-MEC system adopts the partial computation offloading mode, where the task of each UE can be divided into two parts, i.e., one part computed locally at the UE and another part uploaded to the UAV-MEC sever.

Let $\rho_{k,n}$ denote the subchannel assignment indicator, $\forall k \in \mathcal{K}, n \in \mathcal{N}$. Specially, $\rho_{k,n} = 1$ if the k th UE is assigned to the n th subchannel, otherwise $\rho_{k,n} = 0$. According to the uplink NOMA protocol, the received signal at the UAV on the n th subchannel can be expressed as

$$y_n = \sum_{k=1}^K \rho_{k,n} \sqrt{g_{k,n} p_{k,n}} s_{k,n} + z_n, \tag{3}$$

where $s_{k,n}$ represents the transmitted signal of the k th UE over the n th subchannel satisfying $\mathbb{E}\{|s_{k,n}|^2\} = 1$, $p_{k,n}$ is the corresponding transmission power, and z_n stands for the additive white Gaussian noise (AWGN) at the n th subchannel, i.e., $z_n \sim \mathcal{CN}(0, n_0 W_n)$, in which n_0 denotes the single-sided power spectral density of AWGN.

On each subchannel, the SIC decoding order of the uplink NOMA is always from the strong UE to the weak UE so as to avoid consuming significant transmission power for compensating the path loss [24]. Then, the uplink offloading rate of the k th UE on the n th subchannel is given by

$$\begin{aligned}
 R_{k,n} &= \rho_{k,n} W_n \log_2 \left(1 + \frac{g_{k,n} p_{k,n}}{\sum_{i \in S_{k,n}} g_{i,n} \rho_{i,n} p_{i,n} + n_0 W_n} \right) \\
 &= W_n \log_2 \left(1 + \frac{g_{k,n} \rho_{k,n} p_{k,n}}{\sum_{i \in S_{k,n}} g_{i,n} \rho_{i,n} p_{i,n} + n_0 W_n} \right),
 \end{aligned} \tag{4}$$

where $S_{k,n} = \{ \mathcal{K} | g_{i,n} < g_{k,n}, i \neq k \}$ denotes the set of the UEs in the set \mathcal{K} whose channel power gain are worse than that of the k th UE.

Therefore, the number of offloaded bits of the k th UE on the n th subchannel can be calculated as

$$L_{k,n}^{\text{off}} = T R_{k,n}, \tag{5}$$

and the corresponding consumed energy is

$$E_{k,n}^{\text{off}} = \rho_{k,n} \xi_k p_{k,n} T + p_{c,k} T, \tag{6}$$

where ξ is the amplifier coefficient at the UE, and p_c is the fixed circuit power consumption for the computation offloading process at the UE.

According to [19] and [20], the CE of the k th user can be quantified as

$$\begin{aligned}
 \eta_k &= \frac{\sum_{n=1}^N L_{k,n}^{\text{off}} + L_k^{\text{loc}}}{\sum_{n=1}^N E_{k,n}^{\text{off}} + E_k^{\text{loc}}} \\
 &= \frac{R_k}{P_k} = \frac{\sum_{n=1}^N R_{k,n} + \frac{f_k}{C_k}}{\sum_{n=1}^N \rho_{k,n} \xi_k p_{k,n} + \gamma_k f_k^3 + p_{c,k}}.
 \end{aligned} \tag{7}$$

3.4 Problem formulation

With the purpose of guaranteeing fairness for all the UEs, the subcarrier assignment $\{\rho_{k,n}\}$, the local CPU frequency $\{f_{k,n}\}$, and the transmission power $\{p_{k,n}\}$, $\forall k \in \mathcal{K}, n \in \mathcal{N}$, are jointly optimized to maximize the minimum CE among all the UEs. Therefore, the CE optimization problem based on the max-min fairness principle can be formulated as

$$\max_{\{\rho_{k,n}, p_{k,n}, f_{k,n}\}} \min_{k \in \mathcal{K}} \{ \eta_k \} \tag{8a}$$

$$\text{s.t. } \mathcal{C}_1 : R_k \geq R_k^{\min}, \forall k \in \mathcal{K}, \tag{8b}$$

$$\mathcal{C}_2 : P_k \leq P_k^{\max}, \forall k \in \mathcal{K}, \tag{8c}$$

$$\mathcal{C}_3 : 0 \leq f_k \leq f_k^{\max}, \forall k \in \mathcal{K}, \tag{8d}$$

$$\mathcal{C}_4 : \sum_{k=1}^K \rho_{k,n} \leq M_n, \forall n \in \mathcal{N}, \tag{8e}$$

$$\mathcal{C}_5 : \sum_{n=1}^N \rho_{k,n} \leq 1, \forall k \in \mathcal{K}, \tag{8f}$$

$$\mathcal{C}_6 : \rho_{k,n} \in \{0, 1\}, \forall k \in \mathcal{K}, n \in \mathcal{N}, \tag{8g}$$

where R_k^{\min} in the constraint \mathcal{C}_1 is the minimum CB per second required by the k th UE, P_k^{\max} in the constraint \mathcal{C}_2 denotes the maximum power consumption of the k th UE, f_k^{\max} in the constraint \mathcal{C}_3 denotes the maximum CPU frequency of the k th UE, the constraint

\mathcal{C}_4 indicates that the n th subchannel can support at most M_n UEs, and the constraints \mathcal{C}_5 and \mathcal{C}_6 ensure that one UE can occupy at most one subchannel.

4 Proposed computation efficiency optimization algorithm

It is clearly that the problem in (8) belongs to the non-convex MINLP, that is NP-hard to find the global optimal solution in general. In this section, we design a polynomial-time iterative algorithm based on the big-M reformulation, the penalized SCP, and the generalized Dinkelbach's method to obtain a feasible suboptimal solution of (8).

4.1 Big-M reformulation

Due to the product term $\rho_{k,n}p_{k,n}$ in (8), it is not easy to design an efficient algorithm for solving (8). Fortunately, we can use the big-M reformulation to decompose the product terms [25]. With the help of big-M reformulation, we first introduce auxiliary variables $\tilde{p}_{k,n} = \rho_{k,n}p_{k,n}, \forall k \in \mathcal{K}, n \in \mathcal{N}$. Then, all the product terms $\rho_{k,n}p_{k,n}$ in (8) will be replaced with $\tilde{p}_{k,n}$, which yields

$$\tilde{\eta}_k = \frac{\tilde{R}_k}{\tilde{P}_k} = \frac{\sum_{n=1}^N \tilde{R}_{k,n} + \frac{f_k}{C_k}}{\tilde{P}_k}, \tag{9a}$$

$$\tilde{\mathcal{C}}_1 : \tilde{R}_k \geq R_k^{\min}, \forall k \in \mathcal{K}, \tag{9b}$$

$$\tilde{\mathcal{C}}_2 : \tilde{P}_k \leq P_k^{\max}, \forall k \in \mathcal{K}, \tag{9c}$$

where

$$\tilde{R}_{k,n} = W_n \log_2 \left(1 + \frac{g_{k,n} \tilde{p}_{k,n}}{\sum_{i \in S_{k,n}} g_{i,n} \tilde{p}_{i,n} + n_0 W_n} \right), \tag{10a}$$

$$\tilde{P}_k = \sum_{n=1}^N \xi_k \tilde{p}_{k,n} + \gamma_k f_k^3 + p_{c,k}. \tag{10b}$$

Afterwards, we add the following constraints into (8):

$$\mathcal{C}_7 : \tilde{p}_{k,n} \leq P_k^{\max} \rho_{k,n}, \forall k \in \mathcal{K}, n \in \mathcal{N}, \tag{11a}$$

$$\mathcal{C}_8 : \tilde{p}_{k,n} \leq p_{k,n}, \forall k \in \mathcal{K}, n \in \mathcal{N}, \tag{11b}$$

$$\mathcal{C}_9 : \tilde{p}_{k,n} \geq p_{k,n} - (1 - \rho_{k,n}) P_k^{\max}, \forall k \in \mathcal{K}, n \in \mathcal{N}, \tag{11c}$$

$$\mathcal{C}_{10} : \tilde{p}_{k,n} \geq 0, \forall k \in \mathcal{K}, n \in \mathcal{N}. \tag{11d}$$

Moreover, we can rewrite the binary constraint \mathcal{C}_6 in its equivalent continuous form as follows:

$$\mathcal{C}_{6a} : 0 \leq \rho_{kn} \leq 1, \forall k \in \mathcal{K}, n \in \mathcal{N}, \tag{12a}$$

$$\mathcal{C}_{6b} : \sum_{k=1}^K \sum_{n=1}^N (\rho_{k,n} - \rho_{k,n}^2) \leq 0. \tag{12b}$$

Therefore, we can reformulate (8) as the following equivalent problem:

$$\max_{\Theta} \min_{k \in \mathcal{K}} \{\tilde{\eta}_k\} \tag{13}$$

$$\text{s.t. } \tilde{\mathcal{C}}_1, \tilde{\mathcal{C}}_2, \mathcal{C}_3, \mathcal{C}_4, \mathcal{C}_5, \mathcal{C}_{6a}, \mathcal{C}_{6b}, \mathcal{C}_7 \sim \mathcal{C}_{10},$$

where $\Theta = \{\rho_{k,n}, p_{k,n}, \tilde{p}_{k,n}, f_{k,n}, \forall k \in \mathcal{K}, n \in \mathcal{N}\}$.

The big-M reformulation linearizes the term $\rho_{k,n}p_{k,n}$, such that $\rho_{k,n}$ is separated from $p_{k,n}$, which helps to design an efficient CE optimization algorithm. Moreover, if the constraints C_{6a} , C_{6b} , and $C_7 \sim C_{10}$, are satisfied, then \tilde{C}_1 is equivalent to C_1 , and \tilde{C}_2 is also equivalent to C_2 . Hereto, the original MINLP problem in (8) has been equivalently transformed into a more tractable form in (13). However, the transformed problem in (13) still belongs to the non-convex max-min fractional programming (MMFP), which can not be addressed directly. To this end, we combine fractional programming theory with sequential convex programming (SCP) [26, 27].

4.2 Sequential convex programming

In particular, the core idea of SCP is to convert a non-convex optimization problem into a series of convex optimization problems and solve them iteratively, where the non-convex terms in each iteration will be substituted by appropriate inner convex terms [28].

We note that $\tilde{R}_{k,n}$ in (13) can be expressed as the subtraction of two concave functions $\tilde{R}_{k,n,1}$ and $\tilde{R}_{k,n,2}$, i.e.,

$$\begin{aligned} \tilde{R}_{k,n} &= \tilde{R}_{k,n,1} - \tilde{R}_{k,n,2} \\ &= W_n \log_2 \left(\sum_{i \in S_{k,n}} g_{i,n} \tilde{p}_{i,n} + g_{k,n} \tilde{p}_{k,n} + n_0 W_n \right) \\ &\quad - W_n \log_2 \left(\sum_{i \in S_{k,n}} g_{i,n} \tilde{p}_{i,n} + n_0 W_n \right). \end{aligned} \tag{14}$$

At the t th iteration ($t \geq 1$) of SCP, we can obtain an upper bound $\tilde{R}_{k,n,2}^{(t)}$ of $\tilde{R}_{k,n,2}$ by using its first-order Taylor expansion, which is expressed as

$$\begin{aligned} \tilde{R}_{k,n,2} &\leq \tilde{R}_{k,n,2}^{(t)} = W_n \log_2 \left(\sum_{i \in S_{k,n}} g_{i,n} \tilde{p}_{i,n}^{(t-1)} + n_0 W_n \right) \\ &\quad + \frac{W_n \sum_{i \in S_{k,n}} g_{i,n} \left(\tilde{p}_{i,n} - \tilde{p}_{i,n}^{(t-1)} \right)}{\ln 2 \sum_{i \in S_{k,n}} g_{i,n} \tilde{p}_{i,n}^{(t-1)} + n_0 W_n}, \end{aligned} \tag{15}$$

where $\tilde{p}_{i,n}^{(t-1)}$ is the value of $\tilde{p}_{i,n}$ at the $(t-1)$ th iteration of SCP. Similarly, we can obtain a lower upper $\tilde{\rho}_{k,n}^{(t)}$ of $\rho_{k,n}^2$ by using its first-order Taylor expansion, which is expressed as

$$\rho_{k,n}^2 \geq \tilde{\rho}_{k,n}^{(t)} = \left(\rho_{k,n}^{(t-1)} \right)^2 + 2\rho_{k,n}^{(t-1)} \left(\rho_{k,n} - \rho_{k,n}^{(t-1)} \right), \tag{16}$$

where $\rho_{k,n}^{(t-1)}$ is the value of $\rho_{k,n}$ at the $(t-1)$ th iteration of SCP. Furthermore, since the function $\min\{\cdot\}$ is increasing, we can obtain a lower bound $\tilde{\eta}_k^{(t)}$ of $\tilde{\eta}_k$ at the t th iteration of SCP, i.e.,

$$\tilde{\eta}_k \geq \tilde{\eta}_k^{(t)} = \frac{\tilde{R}_k^{(t)}}{\tilde{P}_k} = \frac{\sum_{n=1}^N \left(\tilde{R}_{k,n,1} - \tilde{R}_{k,n,2}^{(t)} \right) + \frac{f_k}{C_k}}{\sum_{n=1}^N \xi_k \tilde{p}_{k,n} + \gamma_k f_k^3 + p_{c,k}}. \tag{17}$$

As a result, a convex approximation of (13) at the t th iteration of SCP can be cast as

$$\begin{aligned}
 & \max_{\Theta} \min_{k \in \mathcal{K}} \left\{ \tilde{\eta}_k^{(t)} \right\} \\
 & \text{s.t. } \tilde{\mathcal{C}}_1 : \sum_{n=1}^N \left(\tilde{R}_{k,n,1} - \tilde{R}_{k,n,2} \right) + \frac{f_k}{C_k} \geq R_k^{\min}, \forall k \in \mathcal{K}, \\
 & \quad \tilde{\mathcal{C}}_2, \mathcal{C}_3, \mathcal{C}_4, \mathcal{C}_5, \mathcal{C}_{6a}, \\
 & \quad \tilde{\mathcal{C}}_{6b} : \sum_{k=1}^K \sum_{n=1}^N \left(\rho_{k,n} - \tilde{\rho}_{k,n}^{(t)} \right) \leq 0, \forall k \in \mathcal{K}, n \in \mathcal{N}, \\
 & \quad \mathcal{C}_7 \sim \mathcal{C}_{10}.
 \end{aligned} \tag{18}$$

4.3 Generalized Dinkelbach's method

It can be observed that all the constraints in (18) are convex, i.e., the feasible region of (18) is a convex set. In addition, for each $\tilde{\eta}_k, \forall k \in \mathcal{K}$, its numerator is non-negative and concave, meanwhile its denominator is positive and convex. Therefore, the problem in (18) belongs to the convex MMFP, which can be efficiently addressed by the generalized Dinkelbach's method [26].

Let λ^* and Θ^* denote the optimal objective and the optimal solution of (18), respectively, then

$$\lambda^* = \max_{\Theta} \min_{k \in \mathcal{K}} \left\{ \frac{\tilde{R}_k^{(t)}(\Theta)}{\tilde{P}_k(\Theta)} \right\} = \min_{k \in \mathcal{K}} \left\{ \frac{\tilde{R}_k^{(t)}(\Theta^*)}{\tilde{P}_k(\Theta^*)} \right\}. \tag{19}$$

Moreover, Θ^* is the optimal solution of (18) if and only if the following condition is satisfied: [26]

$$\begin{aligned}
 & \max_{\Theta} \min_{k \in \mathcal{K}} \left\{ \tilde{R}_k^{(t)}(\Theta) - \lambda^* \tilde{P}_k(\Theta) \right\} \\
 & = \min_{k \in \mathcal{K}} \left\{ \tilde{R}_k^{(t)}(\Theta^*) - \lambda^* \tilde{P}_k(\Theta^*) \right\} \\
 & = 0.
 \end{aligned} \tag{20}$$

Based on (20), we can transform (18) into its associated subtractive form as

$$\begin{aligned}
 & \max_{\Theta} \min_{k \in \mathcal{K}} \left\{ \tilde{R}_k^{(t)} - \lambda^{(q-1)} \tilde{P}_k \right\} \\
 & \text{s.t. } \tilde{\mathcal{C}}_1, \tilde{\mathcal{C}}_2, \mathcal{C}_3, \mathcal{C}_4, \mathcal{C}_5, \mathcal{C}_{6a}, \tilde{\mathcal{C}}_{6b}, \mathcal{C}_7 \sim \mathcal{C}_{10},
 \end{aligned} \tag{21}$$

where (21) is the problem to be solved at the q th iteration ($q \geq 1$) of the generalized Dinkelbach's method, and $\lambda^{(q-1)}$ is the corresponding iterative parameter at the $(q-1)$ iteration. At the q th iteration, we compute the optimal solution $\Theta^{(q)}$ of (21) for the fixed $\lambda^{(q-1)}$, then we update $\lambda^{(q)}$ by using the obtained $\Theta^{(q)}$, i.e.,

$$\lambda^{(q)} = \min_{k \in \mathcal{K}} \left\{ \frac{\tilde{R}_k^{(t)}(\Theta^{(q)})}{\tilde{P}_k(\Theta^{(q)})} \right\}. \tag{22}$$

By solving (21) iteratively and setting a very small iteration tolerance $\epsilon > 0$, ϵ -optimal solution $\Theta^* = \Theta^{(q)}$ of (18) is achieved if the following condition holds: [26]

$$\left| \min_{k \in \mathcal{K}} \left\{ \tilde{R}_k^{(t)}(\Theta^{(q)}) - \lambda^{(q-1)} \tilde{P}_k(\Theta^{(q)}) \right\} \right| \leq \epsilon. \tag{23}$$

Since the objective function in (18) is not smooth, we introduce a new auxiliary variable given as

$$u = \min_{k \in \mathcal{K}} \left\{ \tilde{R}_k^{(t)} - \lambda^{(q-1)} \tilde{P}_k \right\}. \tag{24}$$

Accordingly, the equivalent form of (18) can be expressed as:

$$\begin{aligned} \max_{\Theta, u} \quad & u \\ \text{s.t.} \quad & \bar{C}_1, \bar{C}_2, C_3, C_4, C_5, C_{6a}, \bar{C}_{6b}, C_7 \sim C_{10}, \\ & C_{11} : \tilde{R}_k^{(t)} - \lambda^{(q-1)} \tilde{P}_k \geq u, \forall k \in \mathcal{K}. \end{aligned} \tag{25}$$

Based on the above analysis, a two-layer iterative algorithm consisting of the outer iterative procedure based on the SCP and the inner iterative procedure based on the generalized Dinkelbach’s method can be employed to obtain a feasible suboptimal solution of (13). Nevertheless, the above algorithm requires a feasible point of (13) as the initial point. If the above algorithm starts from a feasible point, then all the generated points will be feasible; otherwise, it may fail at the first iteration due to the infeasibility [28]. However, finding a feasible point in the non-convex feasible region of (13) is very hard. To avoid this, we will integrate the penalized SCP [29] with the framework of the above algorithm, which can start from an arbitrary point.

4.4 Penalized sequential convex programming

To ensure that a sequence of feasible solutions can be generated, the abovementioned SCP needs a feasible initial point. Generally, an infeasible initial point misleads intermediate solutions obtained by the iterative process, which often causes an incorrect local solution. However, it is usually NP hard to find a feasible initial point for non-convex problems, e.g., the CE optimization problem in (13). To avoid the above initialization requirement, it is a good choice to adopt the penalized SCP. Specially, auxiliary variables are introduced to relax all non-convex inequality constraints, and penalty parameters are added into the objective function, so that we can randomly generate the initial point and then feasible solutions can be gradually obtained with the increasing of iterations.

It can be found that only the constraints \bar{C}_1 and C_{6b} are non-convex in (13). To this end, we relax the constraint \bar{C}_1 via slack variables $s_{1,k}$ ($s_{1,k} \geq 0, \forall k \in \mathcal{K}$) and the constraint C_{6b} via a slack variable s_2 ($s_2 \geq 0$). Then, the constraints \bar{C}_1 and \bar{C}_{6b} in (25) are, respectively, reformulated as

$$\check{C}_1 : \sum_{n=1}^N \left(\tilde{R}_{k,n,1} - \tilde{R}_{k,n,2}^{(t)} \right) + \frac{f_k}{C_k} + s_{1,k} \geq R_k^{\min}, \forall k \in \mathcal{K}, \tag{26a}$$

$$\check{C}_{6b} : \sum_{k=1}^K \sum_{n=1}^N \left(\rho_{k,n} - \tilde{\rho}_{k,n}^{(t)} \right) \leq s_2, \forall k \in \mathcal{K}, n \in \mathcal{N}. \tag{26b}$$

In order to minimize the violations of non-convex constraints, we subtract the sum of the slack variables from the objection function of (25). Therefore, (25) can be reformulated as

$$\begin{aligned}
 \max_{\{\Theta, u, s_{1,k}, s_2\}} \quad & u - \tau^{(t-1)} \left(\sum_{k=1}^K s_{1,k} + s_2 \right) \\
 \text{s.t.} \quad & \check{C}_1, \check{C}_2, C_3, C_4, C_5, C_{6a}, \check{C}_{6b}, C_7 \sim C_{11}, \\
 & C_{12} : s_{1,k} \geq 0, \forall k \in \mathcal{K}, \\
 & C_{13} : s_2 \geq 0,
 \end{aligned} \tag{27}$$

where (27) is the problem to be solved at the t th iteration of the penalized SCP, and $\tau^{(t-1)}$ is the corresponding penalty factor at the $(t - 1)$ th iteration and is updated for the next iteration by using the following equation:

$$\tau^{(t)} = \min \left\{ \mu \tau^{(t-1)}, \tau_{\max} \right\}. \tag{28}$$

Here, τ_{\max} denotes the upper bound of $\tau^{(t)}$ so as to avoid numerical problems if $\tau^{(t)}$ is too large and provide convergence if a feasible region is not found, and μ is the predefined increasing factor [29]. Moreover, existing optimization tools, like YALMIP [30], CVX [31], CVXQUAD [32], can be used to solve the convex problem in (27).

In summary, we propose a CE optimization algorithm based on the big-M reformulation, the penalized SCP, and the generalized Dinkelbach’s method, which is presented in Algorithm 1.

Theorem 1 *Algorithm 1 can converge to a local optimal solution of (13).*

Proof See Appendix A. □

Remark 1 *The stopping criterion of Algorithm 1 includes two parts, while one is that the objective value of (13) is nearly unchanged, and the other is that the sum of slack variables is very small, which indicates that the solution obtained by Algorithm 1 is a feasible solution of (13).*

4.5 Computational complexity analysis

In Algorithm 1, the computational complexity is mainly determined by solving problem (27) $L_1 L_2$ times, where L_1 and L_2 denote the required iterations of the inner loop and outer loop, respectively. In addition, (27) is solved by using standard optimization tools based on the interior-point method whose complexity is $\mathcal{O}((4KN + N + 2)^{3.5} \log(1/\varepsilon))$, where $\mathcal{O}(\cdot)$ denotes the big-O notation and ε represents the solution accuracy [33]. Thus, the proposed Algorithm 1 has an acceptable polynomial time computational complexity, i.e., $\mathcal{O}(L_1 L_2 (4KN + N + 2)^{3.5} \log(1/\varepsilon))$.

5 Computing modes as benchmarks

In this section, we discuss the other two computing modes as benchmarks, namely the local computing only mode and the computation offloading only mode. The local computing only mode requires that all the UEs only perform their local computing but never offload their tasks to the UAV-MEC server, while the computation offloading only mode

Algorithm 1 Proposed CE Optimization Algorithm

- 1: **Initialize:** the maximum number of iterations I_1, I_2 , the maximum tolerance ϵ_1, ϵ_2 for inner loop and outer loop, respectively, and an arbitrary point $\{\rho_{k,n}^{(0)}, p_{k,n}^{(0)}, f_{k,n}^{(0)}, \forall k \in \mathcal{K}, n \in \mathcal{N}\}$.
 - 2: Set the iteration index $t = 0$, the iterative objective value $\Gamma^{(0)} = 0$ of (8), the penalty factor $\tau^{(0)} = 1$, the increasing factor $\mu = 1.5$, the upper bound $\tau_{\max} = 10^5$, the sum of penalty values $\pi_1^{(0)} = 0$, and the penalized objective value $\pi_2^{(0)} = 0$ for outer loop.
 - 3: **repeat**
 - 4: Set $t = t + 1$.
 - 5: Set the iteration index $q = 0$, the iterative parameter $\lambda^{(0)} = 0$ for inner loop.
 - 6: **repeat**
 - 7: Set $q = q + 1$.
 - 8: Solve the problem in (27) to obtain the optimal solution $\{\Theta^{(q)}, u^{(q)}, s_{1,k}^{(q)}, s_2^{(q)}, \forall k \in \mathcal{K}\}$.
 - 9: **if** $\left| \min_{k \in \mathcal{K}} \left\{ \tilde{R}_k^{(t)}(\Theta^{(q)}) - \lambda^{(q-1)} \tilde{P}_k(\Theta^{(q)}) \right\} \right| \leq \epsilon_1$ **then**
 Convergence2 = true.
 - else**
 Update $\lambda^{(q)} = \min_{k \in \mathcal{K}} \left\{ \frac{\tilde{R}_k^{(t)}(\Theta^{(q)})}{\tilde{P}_k(\Theta^{(q)})} \right\}$.
 - end if**
 - 10: **until** Convergence2 = true or $q \geq I_2$
 - 11: Update $\Gamma^{(t)} = \min_{k \in \mathcal{K}} \{ \tilde{\eta}_k(\Theta^{(q)}) \}$.
 - 12: Update $\pi_1^{(t)} = \sum_{k=1}^K s_{1,k}^{(q)} + s_2^{(q)}$.
 - 13: Update $\pi_2^{(t)} = \Gamma^{(t)} - \pi_1^{(t)}$.
 - 14: **if** $\pi_1^{(t)} \leq \epsilon_2$ and $|\pi_2^{(t)} - \pi_2^{(t-1)}| \leq \epsilon_2$ **then**
 Convergence1 = true.
 - else**
 Update $\rho_{k,n}^{(t)}, p_{k,n}^{(t)}, f_{k,n}^{(t)}, \forall k \in \mathcal{K}, n \in \mathcal{N}$ according to $\Theta^{(q)}$, and $\tau^{(t)} = \min \{ \mu \tau^{(t-1)}, \tau_{\max} \}$.
 - end if**
 - 15: **until** Convergence1 = true or $t \geq I_1$
 - 16: **Output:** the suboptimal solution of the problem in (8), i.e., $\{\rho_{k,n}^{(t)}, p_{k,n}^{(t)}, f_{k,n}^{(t)}, \forall k \in \mathcal{K}, n \in \mathcal{N}\}$.
-

requires that all the UEs only offload their tasks to the UAV-MEC sever but never perform their local computing.

5.1 Local computing only mode

Under the local computing only mode, the CE of the k th UE can be expressed as

$$\eta_{L,k} = \frac{\frac{f_k}{C_k}}{\gamma k f_k^3 + p_{c,k}}. \tag{29}$$

The corresponding CE optimization problem is formulated as

$$\max_{\{f_k\}} \min_{k \in \mathcal{K}} \{\eta_{L,k}\} \tag{30a}$$

$$\text{s.t. } \frac{f_k}{C_k} \geq R_k^{\min}, \forall k \in \mathcal{K}, \tag{30b}$$

$$\gamma_k f_k^3 + p_{c,k} \leq P_k^{\max}, \forall k \in \mathcal{K}, \tag{30c}$$

$$0 \leq f_k \leq f_k^{\max}, \forall k \in \mathcal{K}. \tag{30d}$$

In (30), we find that the optimization variables $\{f_k, k \in \mathcal{K}\}$ are decoupled. Moreover, the objective function consists of the “min” function and K independent CE expressions, i.e., $\{\eta_{L,k}, k \in \mathcal{K}\}$. Without loss of optimality, we can maximize each CE expression $\eta_{L,k}$ to maximize the objective function of (30). Thus, (30) can be decomposed into K sub-problems, in which the k th sub-problem is given by

$$\max_{f_k} \eta_{L,k} \tag{31a}$$

$$\text{s.t. } f_{L,k}^{\text{LB}} \leq f_k \leq f_{L,k}^{\text{UB}} \tag{31b}$$

$$f_{L,k}^{\text{LB}} = C_k R_k^{\min}, \tag{31c}$$

$$f_{L,k}^{\text{UB}} = \max \left\{ f_k^{\max}, \sqrt[3]{\frac{P_k^{\max} - p_{c,k}}{\gamma_k}} \right\}. \tag{31d}$$

By setting the first order derivative of $\eta_{L,k}$ to be zero, i.e.,

$$\frac{\partial \eta_{L,k}}{\partial f_k} = \frac{-2\gamma_k f_k^3 + p_{c,k}}{C_k (\gamma_k f_k^3 + p_{c,k})^2} = 0, \tag{32}$$

the zero point of $\partial \eta_{L,k} / \partial f_k$ can be calculated as

$$f_{L,k}^{\circ} = \sqrt[3]{\frac{p_{c,k}}{2\gamma_k}}. \tag{33}$$

The numerator of $\partial \eta_{L,k} / \partial f_k$ is a concave and monotone decreasing function with respect to $f_k \geq 0$, while the denominator of $\partial \eta_{L,k} / \partial f_k$ is positive for $f_k \geq 0$. Thus, the closed-form optimal solution of (31) can be derived as

$$f_{L,k}^* = \min \{f_{L,k}^{\text{UB}}, \max \{f_{L,k}^{\text{LB}}, f_{L,k}^{\circ}\}\}. \tag{34}$$

As a result, the best optimal solution of (30) is given by

$$f_L^* = \{f_{L,k}^*, \forall k \in \mathcal{K}\}. \tag{35}$$

Theorem 2 *The problem in (30) has many optimal solutions with the same objective value.*

Proof See [Appendix B](#). □

Remark 2 *Since (30) has many optimal solutions with the same objective value, “the best” means that the optimal solution in (35) can make the CEs of all the UEs as large as possible.*

5.2 Computation offloading only mode

Under the computation offloading only mode, the CE of the k th UE can be expressed as

$$\eta_{C,k} = \frac{\sum_{n=1}^N R_{k,n}}{\sum_{n=1}^N \rho_{k,n} \xi_k p_{k,n} + p_{c,k}}. \tag{36}$$

The corresponding CE optimization problem is formulated as

$$\max_{\{\rho_{k,n}, p_{k,n}\}} \min_{k \in \mathcal{K}} \{\eta_{C,k}\} \tag{37a}$$

$$\text{s.t.} \quad \sum_{n=1}^N R_{k,n} \geq R_k^{\min}, \forall k \in \mathcal{K}, \tag{37b}$$

$$\sum_{n=1}^N \rho_{k,n} \xi_k p_{k,n} + p_{c,k} \leq P_k^{\max}, \forall k \in \mathcal{K}, \tag{37c}$$

$$\sum_{k=1}^K \rho_{k,n} \leq M_n, \forall n \in \mathcal{N}, \tag{37d}$$

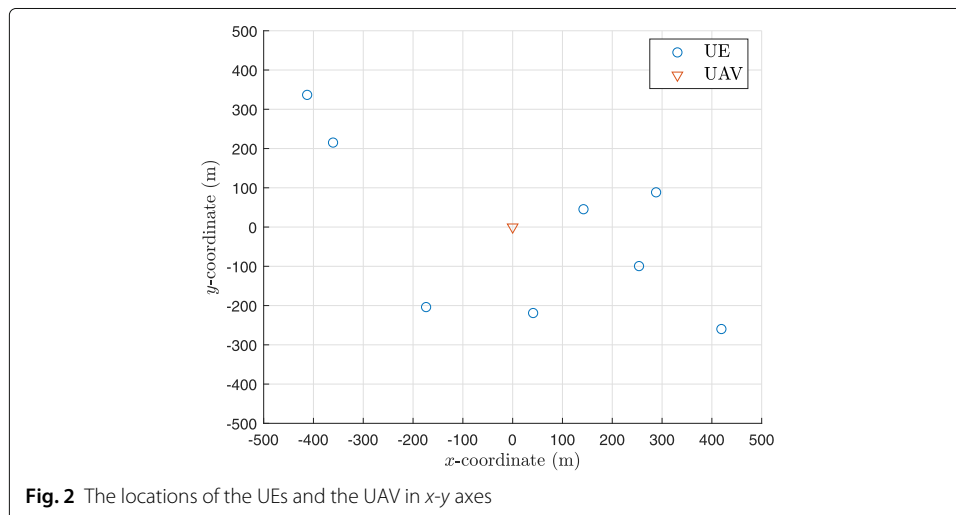
$$\sum_{n=1}^N \rho_{k,n} = 1, \forall k \in \mathcal{K}, \tag{37e}$$

$$\rho_{k,n} \in \{0, 1\}, \forall k \in \mathcal{K}, n \in \mathcal{N}. \tag{37f}$$

Since (37) is similar to (8), the framework of Algorithm 1 can be applied to solve (37). To avoid repetition, the detailed solution to (37) is omitted here.

6 Simulation results and discussions

In this section, we provide the relevant simulation results to first demonstrate the convergence of the proposed Algorithm 1 and then evaluate the performance of the proposed partial offloading scheme by using the local computing only scheme, the computation offloading only scheme, and the OFDMA scheme as benchmarks. In the simulations, we consider a UAV-MEC system as illustrated in Fig. 2, where $K = 8$ UEs are randomly located in a squared area of size 1000 m \times 1000 m, and the UAV is located at



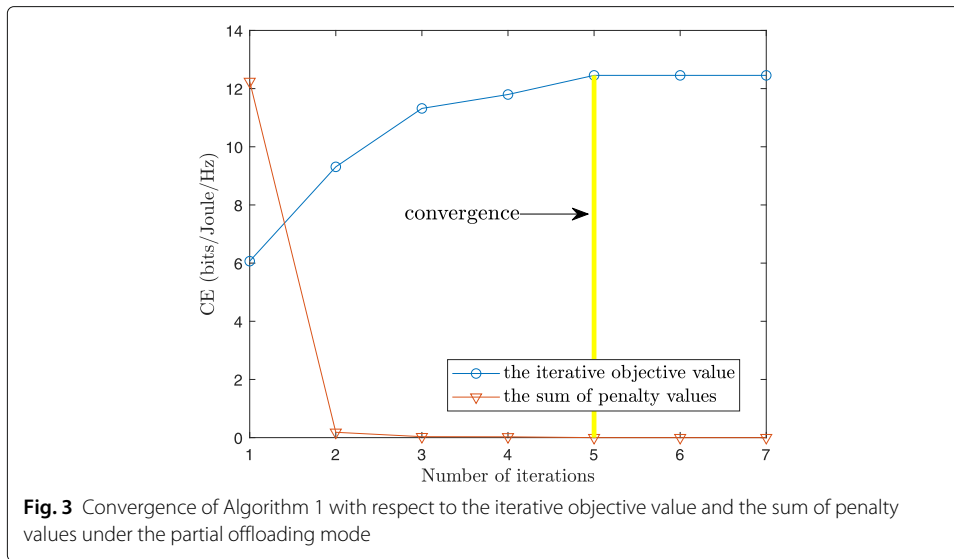
$\mathbf{q}_u = [0, 0, 100]$ [23]. To reduce the decoding complexity and time delay in one NOMA group, the maximum number of the UEs supported by each subchannel is considered as two, i.e., $M_n = 2, \forall n \in \mathcal{N}$ [34]. Besides, the maximum power consumption of the k th UE is $P_k^{\max} = P_{\max}, \forall k \in \mathcal{K}$, where P_{\max} is defined as the maximum local power consumption. For convenience, the normalized CE and CB are used as the performance indicators, and their units are, “Bits/Joule/Hz” and “Bits/Hz”, respectively. Except where otherwise stated, we present the remaining simulation parameters in Table 1, which are based on the typical settings in [21–23].

Figure 3 plots the convergence curves of the iterative objective value $\Gamma^{(t)}$ and the sum of penalty values $\pi_1^{(t)}$ for the proposed Algorithm 1 under the partial offloading mode, where the maximum local power is set as $P_{\max} = 0.2\text{W}$. As seen from Fig. 3, when the number of iterations increases, the sum of penalty values $\pi_1^{(t)}$ decreases accordingly. Since $\pi_1^{(t)}$ denotes the sum of violations of non-convex constraints, the generated solutions by Algorithm 1 tend to be feasible for (8). In fact, Algorithm 1 can be interpreted as an infeasible method for the original MINLP problem in (8). Specially, Algorithm 1 may generate a sequence of infeasible solutions if it starts from an infeasible point but can converge to a feasible suboptimal solution of (8) at last. Moreover, the proposed Algorithm 1 has a fast convergence rate because the iterative objective value $\Gamma^{(t)}$ needs only 5 iterations to be stable and converge to a ϵ -optimal solution for the specified precision ϵ . Furthermore, the average iterations of Algorithm 1’s inner loop and outer loop are 6.08 and 6.49, respectively. The above observations demonstrate the convergence property and the convergence efficiency of Algorithm 1.

Table 2 evaluates the impact of random initial points on the CE performance of Algorithm 1 under the partial offloading mode, where $P_{\max} = 0.2\text{W}$. The random initial points for Algorithm 1 are given by $\rho_{k,n}^{(0)} = \text{rand}([0.5, 1])$, $p_{k,n}^{(0)} = 0, f_{k,n}^{(0)} = 0, \forall k \in \mathcal{K}, n \in \mathcal{N}$, where $\text{rand}([0.5, 1])$ stands for a real random variable uniformly distributed in the range of $[0.5, 1]$. As shown in Table 2, with these random initial points, Algorithm 1 will eventually converge to one of the three different local optimal values, denoted by, $\eta_1 = 10.455$ bits/Joule/Hz with the probability 11%, $\eta_2 = 11.487$ bits/Joule/Hz with the probability 15%, and $\eta_3 = 12.455$ bits/Joule/Hz with the probability 74%, respectively. Moreover, the probability of converging to η_3 is 74% for one run time, which means that Algorithm 1 has a 98.4% probability of converging to the best local optimal values within 3 run times.

Table 1 Simulation parameters

Parameters	Default values
Number of UEs	$K = 8$
Number of subchannels	$N = 4$
Total system bandwidth	$B = 2\text{MHz}$
Reference channel power gain	$\mu_0 = -50\text{dB}$
Noise power density	$N_0 = -150\text{dBm/Hz}$
Required CPU cycles for one bit	$C = 10^3\text{cycles/bit}$
CPU coefficient	$\gamma_k = 10^{-28}$
Amplifier coefficient	$\xi_k = 3$
Fixed circuit power	$p_{c,k} = 50\text{mW}$
Maximum local CPU frequency	$f_k^{\max} = 0.5 \times 10^9\text{cycles/s}$
Minimum required computation bits	$R_k^{\min} = 10^4\text{bits}$
Stop tolerances for Algorithm 1	$\epsilon_1 = \epsilon_2 = 10^{-3}$



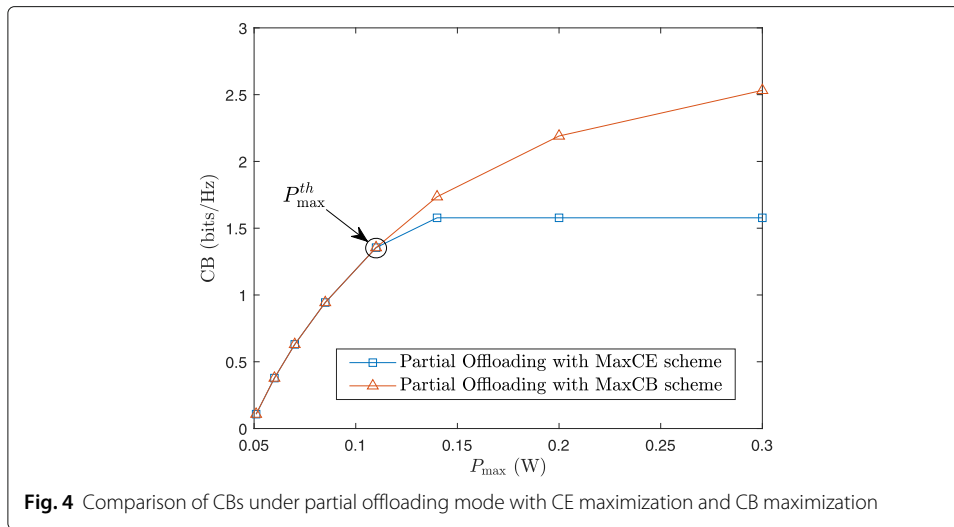
In other words, Algorithm 1 is robust to different initial points due to the high probability of converging to the best local optimal values with very few run times.

In Figs. 4 and 5, we compare the proposed partial offloading mode with the CE maximization scheme referred as “MaxCE” and the CB maximization scheme referred as “MaxCB” in terms of CB and CE with varying the maximum local power consumption P_{max} , respectively. In particular, the CB maximization scheme is to maximize the minimum CB among all the UEs, which is implemented by using the simplified Algorithm 1 without the inner loop, i.e., $\lambda^{(q)} = 0, \forall q$. From the above two figures, the “MaxCE” scheme has both CB and CE in common with the “MaxCB” scheme when P_{max} is not so large, i.e., $P_{max} \leq P_{max}^{th}$. Due to the limitation of P_{max} , the resource allocation strategies of the two schemes are identical when P_{max} is small, which leads to the above result. In the case of small P_{max} , the CE and CB of the two schemes increase with P_{max} since the CB increases faster than the power consumption. But on the other hand, when $P_{max} > P_{max}^{th}$, although the “MaxCB” scheme has larger CB than the “MaxCE” scheme, the “MaxCE” scheme has obvious advantages on the CE compared with the “MaxCB” scheme from Fig. 5. This is because the “MaxCB” scheme maximizing the CB will consume more inefficient power to achieve higher CB but the “MaxCE” scheme will still adopt the solution maximizing the CE instead of consuming inefficient power when $P_{max} > P_{max}^{th}$, i.e., the solution of the “MaxCE” scheme is not affected by the increase of P_{max} . Consequently, both CE and CB of the “MaxCE” scheme keep unchanged when $P_{max} > P_{max}^{th}$.

Figure 6 presents the CE versus the maximum local power consumption P_{max} under different computing modes with the proposed CE maximization frameworks. From Fig. 6, it can be found that the partial offloading mode has a much higher CE performance than the other two computing mode, especially when compared to the local computing only

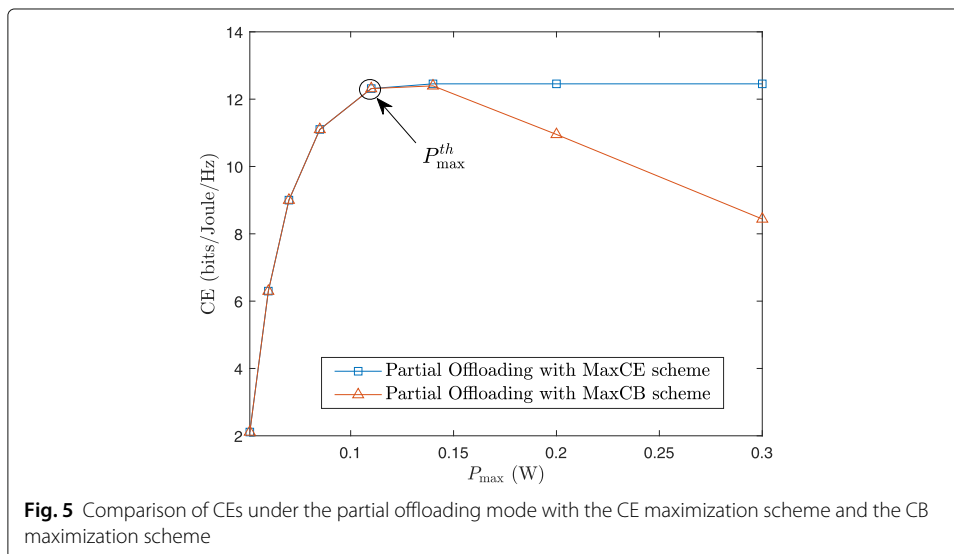
Table 2 Impact of random initial points on Algorithm 1

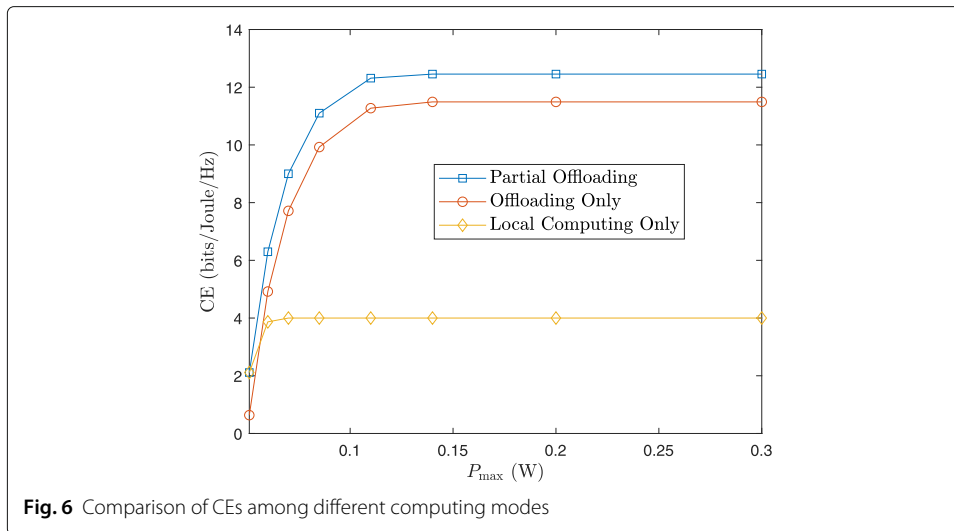
Local optimal objective value	Probability (%)
$\eta_1 = 10.455$ bits/Joule/Hz	11
$\eta_2 = 11.487$ bits/Joule/Hz	15
$\eta_3 = 12.455$ bits/Joule/Hz	74



mode. The reason is that the partial offloading scheme can offer the flexibility to adjust the resource allocation for the task offloading and the local computing. However, the local computing only scheme allows the UEs to perform their local computing only regardless of the channel conditions, and the computation offloading only mode completely uploads the tasks of the UEs to the UAV-MEC server even if the channel conditions are bad, so their CE performances are greatly compromised. Besides, the computation offloading only mode exhibits better CE performance than the local computing only mode since the former can yield better computation-efficient benefits than the latter when P_{\max} is not so small.

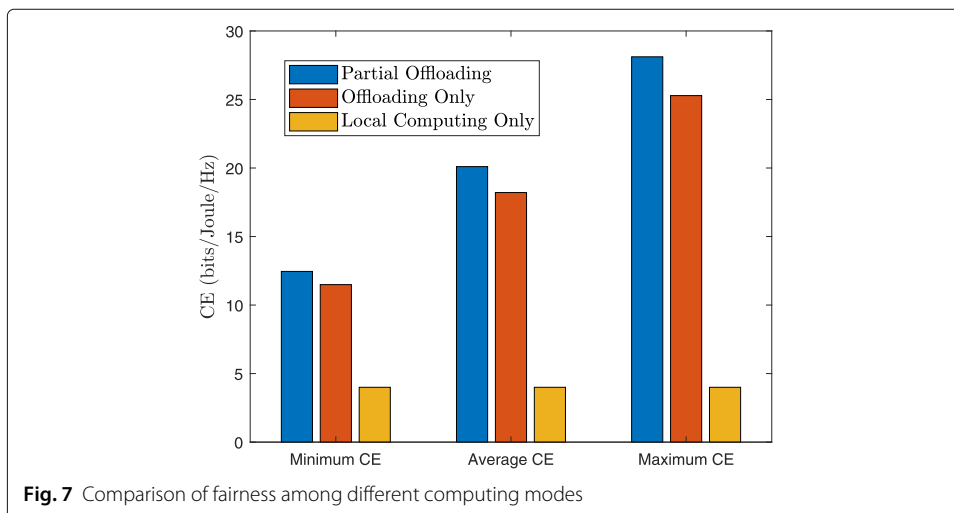
Figure 7 shows the max-min fairness with the proposed CE maximization framework under different computing modes, where the minimum CE, the average CE, and the best CE of all the UEs are compared, and the maximum local power is set as $P_{\max} = 0.2W$. With respect to the above three performance indicators, the proposed partial offloading mode has the best performances, while the local computing only mode has the worst

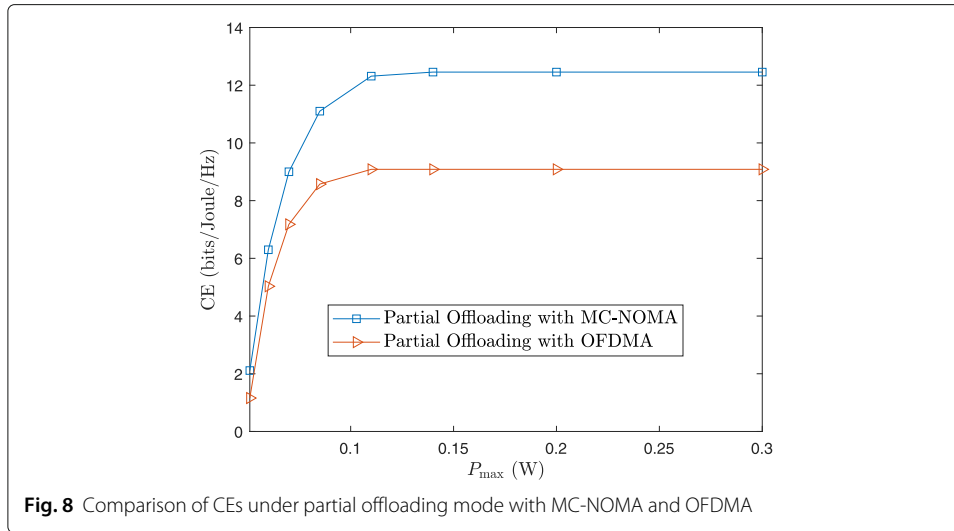




performances. The reason for explaining this is similar to the simulation result in Fig. 6, which verifies the effectiveness of the proposed scheme in the max-min fairness.

Figure 8 illustrates the CE versus the maximum local power consumption P_{\max} in the partial offloading mode with the proposed MC-NOMA scheme and the OFDMA scheme. In the OFDMA scheme, each UE can access one subchannel at most and each subchannel can support one UE at most, which is implemented by letting the maximum number of the UEs supported by one subchannel to be one, i.e., $M_n = 1, \forall n \in \mathcal{N}$. To ensure fair comparison, the number of subchannels in the OFDMA scheme is set as $N = 8$, but the other simulation parameters are consistent with the proposed MC-NOMA scheme. It can be observed from Fig. 8 that the proposed MC-NOMA scheme is significantly greater than the OFDMA scheme on the CE performance. This is because NOMA can service more users than OFDMA at the same frequency-domain resource block, i.e., the multiplexing gain can be obtained when NOMA is applied.





7 Conclusion

In this paper, we have studied a max-min fairness based CE optimization problem in a UAV-MEC system with MC-NOMA, where the joint optimization of the subchannel assignment, the local CPU frequency, and the transmission power is considered. Since it is NP-hard to find the global optimal solution of the formulated non-convex MINLP problem, we design a polynomial-time algorithm to obtain a feasible suboptimal solution. In specify, the big-M reformulation is employed firstly to obtain the equivalent continuous form of the original problem. Next, by applying the penalized SCP and the general Dinkelbach’s method, a two-layer iterative algorithm is designed, which is able to start from an arbitrary initial point and finally converge to a feasible suboptimal solution. We also analyze the corresponding CE optimization problems in the benchmark computing modes. The relevant simulation results have demonstrated that the proposed Algorithm 1 enjoys a good convergence performance. Moreover, the proposed computation-efficient scheme is superior to the other benchmark schemes. Besides, when compared with the conventional OFDMA scheme, the proposed MC-NOMA scheme can get higher CE.

Appendix A

Proof of Theorem 1

First, we prove the convergence of the algorithm based on the SCP and generalized Dinkelbach’s method to solve (13), namely Algorithm 2. Let $\Theta_{(t)}^*$ and $\tilde{\Gamma}^{(t)}(\Theta_{(t)}^*)$ respectively denote the optimal solution and the optimal objective value of (18) at the t th iteration of SCP; meanwhile, the corresponding objective value of (13) is given by $\Gamma(\Theta_{(t)}^*)$, then we have

$$\Gamma(\Theta_{(t)}^*) \stackrel{(a)}{\geq} \tilde{\Gamma}^{(t)}(\Theta_{(t)}^*) \stackrel{(b)}{\geq} \tilde{\Gamma}^{(t)}(\Theta_{(t-1)}^*) \stackrel{(c)}{=} \Gamma(\Theta_{(t-1)}^*). \tag{38}$$

At the t th iteration of SCP, we can find from (38) that (a) holds since $\tilde{\Gamma}^{(t)}(\Theta_{(t)}^*)$ is a lower bound of $\Gamma(\Theta_{(t)}^*)$, (b) holds if (18) can be solved globally, which can be guaranteed by using the generalized Dinkelbach’s method, and (c) holds because $\Theta_{(t-1)}^*$ is used as the local point of the first-order Taylor expansion based on (17). From (38), it can be seen that the objective value of (13) is non-decreasing. Since the objective value of (13) is upper

bounded, Algorithm 2 can gradually converge to a local optimal solution of (13). Next, it is ready to prove the convergence of Algorithm 1. If the penalty factor τ is large enough and the sum of slack variables is very close to zero, then (27) can be rewritten as (25). Therefore, Algorithm 1 is equivalent to Algorithm 2, which indicates that Algorithm 1 will converge to a local optimal of (13). Please refer to [29] for more detail.

Appendix B

Proof of Theorem 2

Based on (35), we denote $\eta_{L,k}(f_{L,k}^*)$ as the optimal CE of the k th UE. We further define \tilde{k} as the index of the UE whose optimal CE is smallest, i.e.,

$$\tilde{k} = \arg \min_{k \in \mathcal{K}} \eta_{L,k}(f_{L,k}^*). \tag{39}$$

According to (39), we have

$$\eta_{L,k}(f_{L,k}^*) \geq \eta_{L,\tilde{k}}(f_{L,\tilde{k}}^*), \forall k \in \mathcal{K} \setminus \tilde{k}, \tag{40}$$

where $\mathcal{K} \setminus \tilde{k} = \{k | k \in \mathcal{K}, k \neq \tilde{k}\}$. For each $k \in \mathcal{K} \setminus \tilde{k}$, there exists at least one $f_k \in [f_{L,k}^{LB}, f_{L,k}^{UB}]$, such that $\eta_{L,k}(f_{L,k}^*) \geq \eta_{L,k}(f_k) \geq \eta_{L,\tilde{k}}(f_{L,\tilde{k}}^*)$. Note that the solution $\{f_k, \forall k \in \mathcal{K} \setminus \tilde{k}, f_{L,\tilde{k}}^*\}$ has the same objective value of (30) as the optimal solution $\{f_{L,k}^*, \forall k \in \mathcal{K}\}$ in (35). Therefore, (30) has many optimal solutions with the same objective value.

Abbreviations

UAV: Unmanned aerial vehicle; MCC: Mobile cloud computing MEC: Mobile edge computing; OMA: Orthogonal multiple access; TDMA: Time division multiple access; OFDMA: Orthogonal frequency division multiple access NOMA: Non-orthogonal multiple access; MC-NOMA: Multi-carrier non-orthogonal multiple access; IoT: Internet of Things; ORB: Orthogonal resource block; SIC: Successive interference cancelation; UE: User equipment; BS: Base station; LoS: Line-of-sight; CB: Computation bits; CE: Computation efficiency; MINLP: Mixed-integer nonlinear programming; SCP: Sequential convex programming; CPU: Central processing unit; 3D: Three-dimension; CSI: Channel state information; AWGN: Additive white Gaussian noise; MMFP: Max-min fractional programming

Acknowledgements

The authors thank the editor and anonymous reviewers for offering their valuable comments, which greatly improve the quality of this paper.

Authors' contributions

FX, YX, JC, and GW conceived and designed the research and simulations. FX designed the main idea and algorithm and drafted the main manuscript of this work. YX gave crucial suggestions and reviewed and modified the drafted manuscript. JC helped to perform the simulation and analyse the result. GW helped to draft the manuscript. All authors reviewed and edited the manuscript. All authors read and approved the final manuscript.

Funding

This work was supported in part by Open Research Fund of Nanjing University of Aeronautics and Astronautics (KFJJ20190409), National Natural Science Foundation of China (61971220) and Open Research Fund Key Laboratory of Wireless Sensor Network and Communication of Chinese Academy of Sciences (2017006), and Postgraduate Research and Practice Innovation Program of Jiangsu Province (KYCX19_0191).

Availability of data and materials

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Competing interests

The authors declare that they have no competing interests.

Received: 5 May 2020 Accepted: 14 August 2020

Published online: 14 September 2020

References

1. H. T. Dinh, C. Lee, D. Niyato, P. Wang, A survey of mobile cloud computing: architecture, applications, and approaches. *Wirel. Commun. Mob. Comput.* **13**(18), 1587–1611 (2013). <https://doi.org/10.1002/wcm.1203>
2. A. U. R. Khan, M. Othman, S. A. Madani, S. U. Khan, A survey of mobile cloud computing application models. *IEEE Commun. Surv. Tutor.* **16**(1), 393–413 (2014). <https://doi.org/10.1109/SURV.2013.062613.00160>

3. X. Liu, M. Jia, X. Zhang, W. Lu, A novel multichannel Internet of Things based on dynamic spectrum sharing in 5G communication. *IEEE Internet Things J.* **6**(4), 5962–5970 (2019). <https://doi.org/10.1109/JIOT.2018.2847731>
4. N. Abbas, Y. Zhang, A. Taherkordi, T. Skeie, Mobile edge computing: a survey. *IEEE Internet Things J.* **5**(1), 450–465 (2018). <https://doi.org/10.1109/JIOT.2017.2750180>
5. Z. Ding, X. Lei, G. K. Karagiannidis, R. Schober, J. Yuan, V. K. Bhargava, A survey on non-orthogonal multiple access for 5G networks: research challenges and future trends. *IEEE J. Sel. Areas Commun.* **35**(10), 2181–2195 (2017). <https://doi.org/10.1109/JSAC.2017.2725519>
6. C. Li, P. Liu, C. Zou, F. Sun, J. M. Cioffi, L. Yang, Spectral-efficient cellular communications with coexistent one- and two-hop transmissions. *IEEE Trans. Veh. Technol.* **65**(8), 6765–6772 (2016). <https://doi.org/10.1109/TVT.2015.2472456>
7. Y. Saito, Y. Kishiyama, A. Benjebbour, T. Nakamura, A. Li, K. Higuchi, in *2013 IEEE 77th Vehicular Technology Conference (VTC Spring)*, Non-orthogonal multiple access (NOMA) for cellular future radio access, (2013), pp. 1–5. <https://doi.org/10.1109/VTCSpring.2013.6692652>
8. X. Liu, X. Zhang, NOMA-based resource allocation for cluster-based cognitive industrial Internet of Things. *IEEE Trans. Ind. Inform.* **16**(8), 5379–5388 (2020). <https://doi.org/10.1109/TII.2019.2947435>
9. A. Kiani, N. Ansari, Edge computing aware NOMA for 5G networks. *IEEE Internet Things J.* **5**(2), 1299–1306 (2018). <https://doi.org/10.1109/JIOT.2018.2796542>
10. Q. Pham, H. T. Nguyen, Z. Han, W. Hwang, Coalitional games for computation offloading in NOMA-enabled multi-access edge computing. *IEEE Trans. Veh. Technol.* **69**(2), 1982–1993 (2020). <https://doi.org/10.1109/TVT.2019.2956224>
11. L. Liu, B. Sun, X. Tan, Y. S. Xiao, D. H. K. Tsang, in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*, Energy-efficient resource allocation and channel assignment for NOMA-based mobile edge computing, (2019), pp. 1–6. <https://doi.org/10.1109/WCNC.2019.8885993>
12. M. Zeng, V. Fodor, in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Energy-efficient resource allocation for NOMA-assisted mobile edge computing, (2018), pp. 1794–1799. <https://doi.org/10.1109/PIMRC.2018.8580984>
13. J. Zhang, L. Zhou, Q. Tang, E. C. Ngai, X. Hu, H. Zhao, J. Wei, Stochastic computation offloading and trajectory scheduling for UAV-assisted mobile edge computing. *IEEE Internet Things J.* **6**(2), 3688–3699 (2019). <https://doi.org/10.1109/JIOT.2018.2890133>
14. F. Zhou, Y. Wu, R. Q. Hu, Y. Qian, Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing systems. *IEEE J. Sel. Areas Commun.* **36**(9), 1927–1941 (2018). <https://doi.org/10.1109/JSAC.2018.2864426>
15. Y. Qian, F. Wang, J. Li, L. Shi, K. Cai, F. Shu, User association and path planning for UAV-aided mobile edge computing with energy restriction. *IEEE Wirel. Commun. Lett.* **8**(5), 1312–1315 (2019). <https://doi.org/10.1109/LWC.2019.2913843>
16. Q. Liu, L. Shi, L. Sun, J. Li, M. Ding, F. Shu, Path planning for UAV-mounted mobile edge computing with deep reinforcement learning. *IEEE Trans. Veh. Technol.* **69**(5), 5723–5728 (2020). <https://doi.org/10.1109/TVT.2020.2982508>
17. Z. Yang, C. Pan, K. Wang, M. Shikh-Bahaei, Energy efficient resource allocation in UAV-enabled mobile edge computing networks. *IEEE Trans. Wirel. Commun.* **18**(9), 4576–4589 (2019). <https://doi.org/10.1109/TWC.2019.2927313>
18. X. Diao, J. Zheng, Y. Cai, Y. Wu, A. Anpalagan, Fair data allocation and trajectory optimization for UAV-assisted mobile edge computing. *IEEE Commun. Lett.* **23**(12), 2357–2361 (2019). <https://doi.org/10.1109/LCOMM.2019.2943461>
19. F. Zhou, H. Sun, Z. Chu, R. Q. Hu, in *2018 IEEE Global Communications Conference (GLOBECOM)*, Computation efficiency maximization for wireless-powered mobile edge computing, (2018), pp. 1–6. <https://doi.org/10.1109/GLOCOM.2018.8647509>
20. H. Sun, F. Zhou, R. Q. Hu, Joint offloading and computation energy efficiency maximization in a mobile edge computing system. *IEEE Trans. Veh. Technol.* **68**(3), 3052–3056 (2019). <https://doi.org/10.1109/TVT.2019.2893094>
21. F. Zhou, Y. Wu, R. Q. Hu, Y. Qian, in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Computation efficiency in a wireless-powered mobile edge computing network with NOMA, (2019), pp. 1–7. <https://doi.org/10.1109/ICC.2019.8761172>
22. Y. Wu, Y. Wang, F. Zhou, R. Qingyang Hu, Computation efficiency maximization in OFDMA-based mobile edge computing networks. *IEEE Commun. Lett.* **24**(1), 159–163 (2020). <https://doi.org/10.1109/LCOMM.2019.2950013>
23. J. Zhang, L. Zhou, F. Zhou, B. Seet, H. Zhang, Z. Cai, J. Wei, Computation-efficient offloading and trajectory scheduling for multi-UAV assisted mobile edge computing. *IEEE Trans. Veh. Technol.* **69**(2), 2114–2125 (2020). <https://doi.org/10.1109/TVT.2019.2960103>
24. Z. Yang, Z. Ding, P. Fan, N. Al-Dhahir, A general power allocation scheme to guarantee quality of service in downlink and uplink NOMA systems. *IEEE Trans. Wirel. Commun.* **15**(11), 7244–7257 (2016). <https://doi.org/10.1109/TWC.2016.2599521>
25. J. Lee, S. Leyffer (eds.), *Mixed integer nonlinear programming, vol. 154* (The IMA Volumes in Mathematics and its Applications, New York, 2011)
26. A. Zappone, E. Jorswieck, Energy efficiency in wireless networks via fractional programming theory. *Found. Trends Commun. Inf. Theory.* **11**(3–4), 185–396 (2015). <https://doi.org/10.1561/01000000088>
27. A. Zappone, L. Sanguinetti, G. Bacci, E. Jorswieck, M. Debbah, Energy-efficient power control: a look at 5G wireless technologies. *IEEE Trans. Signal Process.* **64**(7), 1668–1683 (2016). <https://doi.org/10.1109/TSP.2015.2500200>
28. B. R. Marks, G. P. Wright, A general inner approximation algorithm for nonconvex mathematical programs. *Oper. Res.* **26**(4), 681–683 (1978)
29. T. Lipp, S. Boyd, Variations and extension of the convex–concave procedure. *Optim. Eng.* **17**(2), 263–287 (2016)
30. J. Lofberg, in *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*, Yalmip : a toolbox for modeling and optimization in MATLAB, (2004), pp. 284–289. <https://doi.org/10.1109/CACSD.2004.1393890>
31. M. Grant, S. Boyd, CVX Research, CVX: Matlab software for disciplined convex programming, version 2.1. Online, (2018). Available: <http://cvxr.com/cvx>
32. H. Fawzi, J. Saunderson, P. A. Parrilo, Semidefinite approximations of the matrix logarithm. *Found. Comput. Math.* **19**(2), 259–296 (2019). Package cvxquad at <https://github.com/hfawzi/cvxquad>

33. A. Ben-Tal, A. Nemirovski, *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*, vol. 2. (Siam, Philadelphia, 2001)
34. D. Zhai, R. Zhang, Joint admission control and resource allocation for multi-carrier uplink NOMA networks. *IEEE Wirel. Commun. Lett.* **7**(6), 922–925 (2018). <https://doi.org/10.1109/LWC.2018.2840723>

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
