Ding *et al. Journal of Cloud Computing* (2023) 12:142
https://doi.org/10.1186/s13677-023-00497-5

Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

Resource allocation strategy for blockchain-enabled NOMA-based MEC networks

Jianjie Ding[1*], Lina Han[1], Jie Li[2] and Dajun Zhang[3]

Abstract

Blockchain technology is getting more and more attention due to its decentralization, independence and security features. However, in wireless networks it faces a computational challenge: the proof-of-work problem. Mobile edge computing (MEC) leads to a vaild scheme by providing cloud computing capabilities to mobile devices. Non-orthogonal multiple access (NOMA) exploits the diversity properties in the power domain to further increase system throughput and spectral efficiency. In this paper, we suggest a new NOMA-based MEC wireless blockchain network to minimize system energy consumption through task offloading decision optimization, user clustering, computing resource and transmit power allocation. In order to effectively figure out this non-convex problem, we first propose a offloading decision and user clustering algorithm, and then propose a computing resource allocation algorithm based on user Quality of Service (QoS) requirements. Finally, the transmission power can be easily determined. The numerical simulation results verify that the proposed joint optimization algorithm can effectively decrease the system energy consumption.

Keywords Mobile edge computing, Non-orthogonal multiple access, Resource allocation, Power control

Introduction

As the wide popularity of various smart devices and complex operations in the network, smart devices have more and more demands for communication and computing resources. The majority of these applications are computationally intensive and sluggishly sensitive [1, 2], they will incur significant latency and energy consumption during runtime. Mobile edge computing (MEC) transfers the operating and storage capacity close to the users, allowing the users to load work tasks to the MEC server for processing. By taking advantage of the vast computing resources owned by edge servers, MEC can bring many benefits to people's lives, such as saving energy consumption for smart devices and reducing computational latency for tasks [3, 4]. It also avoids network congestion of traditional cloud computing [5]. Therefore, many researchers are dedicated to studying the application of MEC in different applications. However, when data transmission is invoked over the wireless link, the performance of MEC is related to the way wireless resources are allocated during data transmission and the amount of computation offloaded by smart devices. Furthermore, to enhance the productivity of users in unloading tasks in the MEC system further, a new hybrid MEC technology is introduced in which intelligent devices can simultaneously offload computing workloads to multiple edge servers using different radio access networks [6, 7].

In MEC systems, it is not enough to achieve low delay and low energy consumption for system performance improvement, the spectral efficiency of the system is also

*Correspondence:
Jianjie Ding
dingjianjie@yeah.net
[1] School of Mathematics and Statistic, Shaanxi XueQian Normal University, Xi'an 710121, China
[2] School of Physical Education, Xi'an Fanyi University, Xi'an 710105, China
[3] School of Information Technology, Carleton University, Ottawa ON, Canada

an essential consideration [8]. Non-orthogonal multiple access (NOMA) [9, 10] as a prospective technology nowadays. NOMA allows two or more users to jointly use a channel resource or time slot to transmit information, and uses continuous interference cancellation at the receiver end to mitigate common channel interference from user devices [11], which can enhance the efficiency of system spectrum usage [12]. The potential benefits of NOMA have been demonstrated by a number of efforts, such as as enhanced system capacity, energy utilization and frequency spectrum utilization. Therefore, consider combining NOMA with MEC to expand the number of users in the network and improve spectrum utilization [13, 14]. Despite NOMA has many advantages, it still requires careful management of wireless resource allocation. For example, power allocation to reduce co-channel interference for users within the NOMA cluster, which has attracted a great deal of research work.

In addition, it is very important to guarantee the accuracy and reliability of billing services during the transmission of user data. However, fee information is vulnerable to disclosure or artificial manipulation. Attackers gain access to users by tracking their paid information, location privacy and other information [15]. Blockchain technology is a very promising distributed technology that can be a good solution to solve the privacy and security in the network. It has the features of anonymity, security and is convenient for information interconnection [16]. And it can be used in a large range of usable areas in power markets and other energy trading scenarios. For example, in [17], the authors design an energy auction system based on smart contract for users to choose safe and free power transactions. In [18], federated blockchains are used to set up multi-agent nodes through which users' choice of charging modes and transactions can be managed. Although the emergence of blockchain has greatly helped many existing efforts, the consumption of computing resources in the process of blockchain consensus is still an issues that need to be addressed urgently. Therefore, applying blockchain to MEC can improve the security of user information in MEC system during transmission. Meanwhile, MEC can be a good solution to the problem of shortage of node computing resources in blockchain.

In this paper, we discuss a system integrating NOMA and MEC for data delivery by two users over the same channel, which leads to interference between users. And use blockchain as a tool for task unload solutions under the MEC architecture. For the purpose of lessening the impact of such disturbances on system energy consumption, we propose a problem to minimize the total energy consumption of the system by jointly optimizing user offloading decisions, user pairing, computing resource division and user's sending power. This problem is extremely difficult to solve. In order to solve this question, we propose a heuristic algorithm with low complexity to solve it. We first solve for the two integer variables of user offloading decisions and user clustering, and then we allocate power and computational resources in the direction of making the system energy consumption reduced while ensuring that the maximum system latency is satisfied. The validation of the simulation part illustrates the proposed method has obvious advantages over other baseline methods.

The main content of each remaining section of the article is summarized below. In Releated works section, we introduce the research status of the system combined with NOMA and MEC and the research work on the combination of MEC and blockchain respectively. In System model section, we describe the model of the system and optimization problem. In Offloading decision and user clustering optimization section, we express the solution of offloading decision and user clustering. In Computing resource block allocation section, we present the method of computing resource division. In Transmit power control section, we propose the method of user power distribution. The simulation results of the article are presented in Simulation results section. Finally, the main points of the paper are summarized in Conclusions section.

Releated works

In recent years, blockchain has a number of merits, such as decentralization, invariance and transparency, which can handle the security issues faced by edge computing [19]. In addition, blockchain is also critical to improving performance in other domains, such as intelligent healthcare, surveillance networks, smart cities, and the Internet of Things [20]. While blockchain is easy to use in other areas, it also has a major disadvantage, and can be scaled to a small extent. Therefore, its application in the above aspects still has some limitations. For enhancing the functionality of blockchain, edge computing data can be leveraged to scale it. In [21], for the purpose of enhancing the security and stability of cache management, the authors design a new edge cache scheme based on block chain. [22] proposes an incentive scheme for blockchain-based video flow that makes video conversion and user collaboration more secure and reliable. In [23], the authors propose a service-oriented blockchain system architecture to make task offloading between boundary servers more secure in the MEC environment.

The authors of [24] presents a non-trustworthy MEC work verification scheme based on mobile blockchain and design a one-time sorting algorithm for this scheme to more equitably supply computing resources to all IoT mobile users. In [25], the authors develop a framework for collaborative computing offloading and resource partitioning in blockchain and MEC combined systems [26] proposed a task offloading model combining MEC and blockchain, which can guarantee user privacy security while reducing offloading time and energy consumption cost. In [27] , the authors propose a secure unloading framework based on block chain to save the energy consumption of network equipment with low delay cost. However, the majority of these efforts use inefficient consensus mechanisms, ignoring the consideration of throughput issues. In this case, important information is easily leaked, which will threaten the security performance of the system.

As the core key technology of 5G, NOMA can meet the requirements of low cost, low latency and less power consumption of massive connections in the system. Therefore, the integration of NOMA technology and MEC may dramatically improve the big data storage and data transmission rate of the internet of things. Nowadays, many researchers have investigated NOMA-based MEC systems from different perspectives. In [28], the authors jointly optimize transmit energy beamforming, transmit power, and time allocation to maximize the computational efficiency of strong users. In [29], the authors propose a general hybrid offloading of NOMA-MEC systems that minimizes the total energy consumption of the model. The authors of [30] utilize NOMA to achieve large-scale connectivity, which improves the energy efficiency of offloading by optimizing the problem of associating wireless and computing resources [31] combines MEC with NOMA technology to divide the resources of NOMA-MEC system in ultra-dense networks and optimize them effectively. In [32], the problem of joint work unloading and resource partitioning in NOMA-HetNets is investigated to make the energy usage of the system model even lower. In [33], the authors consider the offloading requirements of users near and far, and proposes a delay-aware offloading algorithm based on NOMA cooperative MEC network.

Encouraged by the above research, it can reflect that the user's offloading decision [34] and resource division [35, 36] are very important for the feature optimization of the NOMA-based MEC system. This paper proposes a NOMA-based MEC wireless blockchain network to minimize system energy consumption through optimizing offload forms, user pairing, computing resource partitioning and power partitioning. And the performance of the system is advanced by these strategies.

System model

Consider a scenario where NOMA-assisted MEC wireless blockchain network, there are U users in the system and base stations (BSs) provides task offloading service by wirelessly connecting to MEC server. The blockchain is taken as a non-centralized database to record trade performed in the network and to manage data security between mobile terminal and MEC server. Figure 1 shows the vehicle network structure of blockchain and MEC. The model diagram mainly consists of two parts, device layer and edge layer respectively. The users are defined as $\mathcal{U} = \{1, 2, ..., U\}$, each user can be a miner to record transactions executed in the network through a blockchain application, and that each miner has an indivisible task, each task represented by $\Lambda_u = \{L_u, C_u, T_u^{max}\}$, where L_u is the input-data size (in bits), C_u is the size of

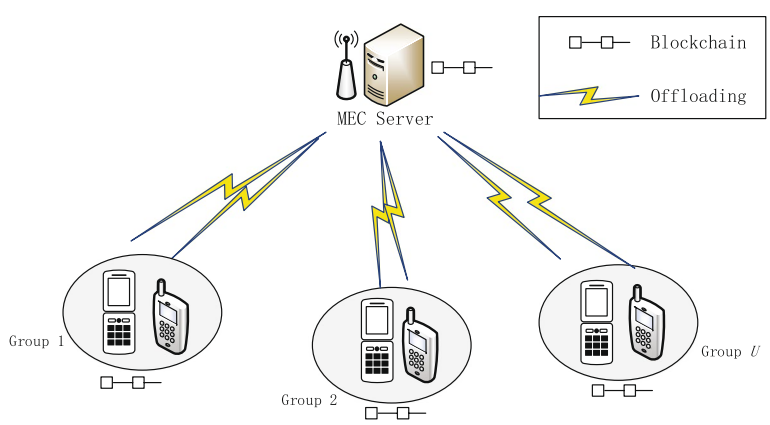

Fig. 1 System model

the workload (in CPU cycles per bit), T_u^{max} is the maximum delay. In addition, each NOMA group will be distributed a quadrature channel with bandwidth B. Tasks can be binary offloaded, i.e., computing tasks cannot be divided into multiple parts and must be processed locally or completely offloaded to the MEC server. We denote it by x_u, $x_u = 0$ means the task is executed locally and $x_u = 1$ represents the work is transferred to the MEC for execution. The mainly used notations are presented in Table 1.

Local model

When tasks can can execute itself locally, define the processing capability of user u in local is f_u^{max}, the execution time T_u^{loc} is

$$T_u^{loc} = \frac{C_u}{f_u^{max}}, \tag{1}$$

The energy consumption E_u^{loc} is given by

$$E_u^{loc} = \alpha C_u f_u^{max2}, \tag{2}$$

where α is a coefficient depending on chip architecture.

MEC model

When the users is unloaded to the MEC server, the users are divided into M groups, the groups are orthogonal and the users in the group share a channel using NOMA. Denote $\mathcal{M} = \{1, 2, ..., M\}$ as the set of groups, and the number of acceptable users in each group is K. $\beta_u^{m,k}$, $\forall m \in \mathcal{M}$, $\forall k \in \mathcal{K}$ as the binary variable to indicate the allocation of user u to kth order of NOMA group m. Here, if $\beta_u^{m,k} = 1$ is allocated to NOMA group m, and $\beta_u^{m,k} = 0$ otherwise. Define h_u as the channel gain for user u transmission task. In NOMA uplink, users with high channel gain are allocated the maximum power possible. Therefore, to effectively apply SIC to decode signals, we assume that the user channel gain is ranked as

$$h_1 > h_2 > ... > h_u, \ \forall u \in \mathcal{U}, \tag{3}$$

Users in each group share the same frequency resources, resulting in interference. Consider interference signals from other users in the group, the signal to interference plus noise ratio (SINR) of the user u in group m can be expressed as

$$\gamma_u = \frac{p_u^{mec} h_u}{\sum\limits_{h_j < h_u} p_j^{mec} h_j + \sigma^2}, \tag{4}$$

where p_u^{mec} is the sending power of user u in the group m, h_u is the channel gain of the user u in the group m, p_j^{mec} is the transmit power of user j in the group m, h_j is the channel gain of the user j in the group m. σ^2 is the noise power. So the task transmit rate of user u in the group m is given by

$$R_u = \sum_{\forall m \in \mathcal{M}} \sum_{\forall k \in \mathcal{K}} \beta_u^{m,k} B \log_2 \left(1 + \frac{p_u^{mec} h_u}{\sum\limits_{h_j < h_u} p_j^{mec} h_j + \sigma^2} \right), \tag{5}$$

Based on R_u, the transmit delay and the energy consumption of user u in the group m are

$$T_u^{tra} = \frac{L_u}{R_u}, \tag{6}$$

$$E_u^{tra} = \frac{L_u}{R_u} p_u^{mec}. \tag{7}$$

When the task is sent to the MEC for processing, there will be latency. Compute resources in each MEC server are provided in the form of compute resource blocks (CRBs), and the ability to deal with tasks of each CRB is f_{unit} CPU cycles per second. Therefore, the task processing delay of user u during this process is

$$T_u^{pro} = \frac{C_u}{f_{unit} \alpha_u^m}, \tag{8}$$

$$E_u^{wait} = \frac{C_u}{f_{unit} \alpha_u^m} p_u^{idle}, \tag{9}$$

where p_u^{idle} is the power of user u when waiting, and α_u^m represents the number of computing resource blocks allocated to user u by group m. So the total processing delay and total energy consumption of the user task u is

Table 1 Notation definitions

Symbol	Definition
L_u	The input data size of the task u
C_u	Size of the workload of user u
f_u^{max}	The processing capability of user u in local
T_u^{max}	Maximum delay of task u
B	The bandwidth of each subchannel
$\beta_u^{m,k}$	The user u is grouped into the kth cluster of NOMA group m
γ_u	The SINR of user u in NOMA group m
p_u^{mec}	The power of user u
h_u	The channel gain of the user u
σ^2	The noise power
f_{unit}	The processing capacity of each CRB
p_u^{idle}	The power consumption of user u when waiting
α_u^m	The number of computing resource blocks in MEC processing allocated to user u by NOMA group m
x_u	The offloading decisions of task u

$$T_u^{mec} = \beta_u^{m,k}(T_u^{tra} + T_u^{pro}), \tag{10}$$

$$E_u^{mec} = \beta_u^{m,k}(E_u^{tra} + E_u^{wait}). \tag{11}$$

System delay and energy consumption

Based on the latency and energy consumption between local processing and MEC processing, the latency of user u is

$$T_u = x_u(T_u^{loc} + T_u^{mec}), \tag{12}$$

The energy consumption of user u is defined as

$$E_u = x_u(E_u^{loc} + E_u^{mec}). \tag{13}$$

Problem formulation

We developed a strategy to minimize the total system energy consumption by optimizing task offloading decision $\Pi = \{x_u\}$, user clustering $\mathbf{B} = \{\beta_u^{m,k}\}$, computing resources $\mathbf{A} = \{\alpha_u^m\}$ and power allocation $\mathbf{P} = \{p_u^{mec}\}$. Specifically, a user set is a group of users who need to complete a communication task, and these users are regarded as a set of users. On the other hand, the clustering representation is based on the NOMA as the encoding technology in our study, so the SIC technology is adopted for decoding. In the decoding, all users in the user set need to be assigned to different clusters, and the subcarrier is shared in each cluster. The optimization problem is formulated as

$$\begin{aligned}
(\boldsymbol{\mathcal{P}_1}): \ & \min_{\Pi,\mathbf{B},\mathbf{A},\mathbf{P}} \sum_{u\in\mathcal{U}} E_u \\
\text{s.t. (C1)}: \ & \mathrm{T_u} \le \mathrm{T_u^{max}}, \ \forall \mathrm{u} \in \mathcal{U}, \\
\text{(C2)}: \ & 0 \le p_u^{mec} \le p_u^{max}, \ \forall u \in \mathcal{U}, \\
\text{(C3)}: \ & x_u \in \{0,1\}, \ \forall u \in \mathcal{U}, \\
\text{(C4)}: \ & \alpha_u^m \in \mathbb{Z}, \ \forall m \in \mathcal{M}, \ \forall u \in \mathcal{U}, \\
\text{(C5)}: \ & \beta_u^{m,k} \in \{0,1\}, \ \forall u \in \mathcal{U}, \ \forall m \in \mathcal{M}, \ \forall k \in \mathcal{K}.
\end{aligned} \tag{14}$$

Where (C1) is the maximum completion delay, T_u represents the maximum assignable power of user u, and T_u^{max} indicates the maximum tolerable delay; (C2) is that the transmitting power should be less than its maximum value, Where p_u^{max} represents the maximum assignable power of user u; (C3) indicates that each user can only have one uninstall option; (C4) is the integer constraints on computational resource blocks devide in MEC servers;(C5) requires each user can only be assigned in one way. Problem ($\boldsymbol{\mathcal{P}_1}$) is a mixed integer nonlinear programming problem, so we decouple it to solve multiple subproblems and developed a heuristic algorithm for solving.

Offloading decision and user clustering optimization

Offloading decision

Since the optimization goal modeled in this paper minimizes system energy consumption. Therefore, we can minimize system energy consumption by offloading decision optimization. We propose a heuristic algorithm to get the user to uninstall the choice of the way $\Pi = \{x_u\}, u \in \mathcal{U}$. Our main idea is summarized as: according to the maximum tolerable delay T_u^{max} to determine where the user task is suitable for processing. The main steps are as follows

(i) If $\frac{C_u}{f_u^{max}} < T_u^{max}$, task is processed locally, we divide the user into the local set $\mathcal{N}_{loc}$.

(ii) If $\frac{C_u}{f_u^{max}} > T_u^{max}$, task is offloaded to MEC processing server, and we divide the user into the MEC set $\mathcal{N}_{mec}$.

Through the above two steps, the solution of binary variable unloading decision is obtained. The specific steps are summarized in lines 1-9 in Algorithm 1.

1: **Initialization**: Set $\mathcal{U}_{loc} = \emptyset, \mathcal{U}_{mec} = \emptyset, \mathcal{U}_{temp} = \mathcal{U}$, $U_{loc} = 0$, $U_{mec} = 0$
2: **Offloading Decision**
3: **for** $u \in \mathcal{U}$ **do**
4: **if** $\frac{C_u}{f_u^{max}} \le T_u^{max}$ **then**
5: $\mathcal{U}_{loc} = \mathcal{U}_{loc} \bigcup u$, $U_{loc} = U_{loc} + 1$
6: **else**
7: $\mathcal{U}_{mec} = \mathcal{U}_{mec} \bigcup u$, $U_{mec} = U_{mec} + 1$
8: **end if**
9: **end for**
10: **User Clustering**
11: **for** each $\mathcal{G}_m$, $m \in \mathcal{M}$ **do**
12: Sort the users in $\mathcal{N}_{mec}$ such that $h_1 \ge h_2 \ge ...h_J \ge ... \ge h_{|\mathcal{G}_m|}$
13: Users {1,2,...,J} is assigned to cluster {1,2,...,J}
14: Users $\{J+1, J+2, ..., J+J\}$ is assigned to cluster $\{1, 2, ..., J\}$...
15: **end for**
16: **Output**: Π and $\mathbf{B}$.

Algorithm 1 Heuristic Offloading Decision and User Clustering Optimization Algorithm

User clustering

According to the NOMA uplink principle and SIC demodulation principle, the greater the channel gain gap, the smaller the interference between users. Therefore, we propose a heuristic algorithm to divide two users into a cluster. The allocation principle is as follows: select users with large channel gain differences as possible as a group. For example, if the system has ten users and the channel gain is sorted in descending order as $h_1 > h_2 > h_3 > ... > h_6 > h_7 > h_8 > h_9 > h_{10}$,

it will be divided into five clusters: user 1 and user 6 in the first cluster, user 2 and user 7 in the second cluster, user 3 and user 8 in the third cluster, user 4 and user 9 in the fourth cluster, user 5 and user 10 in the fifth cluster. The specific user clustering diagram is shown in Fig. 2. See lines 10-15 of algorithm 1 for detailed user assignment steps.

Computing resource block allocation

After the offloading decision and user clustering, computing resources need to be allocated to users, and the original optimization problem ($\mathcal{P}_1$) will be transformed into

$$\begin{aligned} (\mathcal{P}_2): \ & \min_{\mathbf{A}} \sum_{u\in\mathcal{U}} E_u \\ \text{s.t. (C1)}: \ & \mathrm{T_u} \le \mathrm{T_u^{max}}, \ \forall \mathrm{u} \in \mathcal{U}, \\ \text{(C4)}: \ & \alpha_u^m \in \mathbb{Z}, \ \forall m \in \mathcal{M}, \ \forall u \in \mathcal{U}, \end{aligned} \tag{15}$$

Assuming there are adequate computing sources in the MEC to serve the users. When allocating computing resource blocks, each user wants more computing resource blocks to complete their tasks, but for the system, different computing resource blocks allocated to different users will bring different energy gains. Therefore, we perform two rounds of computing resource block allocation:

(i) Allocate a number of computing resource blocks to each user to ensure that the user's minimum latency requirement is met.

(ii) We define the delay gain for a single user: $gain_u^c = T_u^{mec,\alpha_u} - T_u^{mec,\alpha_u+1}$, where α_u denotes the current amount of computing resource blocks allocated to user u, T_u^{mec,α_u} and T_u^{mec,α_u+1} are the delays when user u is allocated with α_u and $\alpha_u + 1$ computing resource block, respectively. For the remaining blocks of computing resources, the user with the largest user delay gain is chosen as the computing resource during each iteration. As a result, the whole procedure of distribution is carried out in the direction of decreasing energy consumption gain. Based on the above idea, our computing resource allocation algorithm is detailed in Algorithm 2.

1: **Initialization**:
2: Set $\mathcal{G}_1 = \emptyset, ..., \mathcal{G}_M = \emptyset$, $G_1 = 0, ..., G_M = 0$,
3: Initialization: $\alpha_u = 0, \ u \in \mathcal{G}_m$
4: **for** $u \in \mathcal{U}$ **do**
5: **repeat**
6: $\alpha_u = \alpha_u + 1$
7: $U_m = U_m - 1$
8: **until** condition $T_u^{up} + T_u^{pro} < T_u^{max}$ is satisfied
9: **end for**
10: **repeat**
11: Calculate $gain_u^c, \forall u \in \mathcal{G}_m$
12: $u^* = \text{argmax}(\{gain_u^c\})$, $\alpha_{u^*} = \alpha_{u^*} + 1$, $U_m = U_m - 1$
13: **until** $U_m = 0$
14: **Output**: $\mathbf{A}$.

Algorithm 2 Heuristic Computing Resource Block Allocation Optimization Algorithm

Transmit power control

After the offloading decision and the solution of MEC user clustering are obtained, the optimization problem ($\mathcal{P}_1$) will degenerate into the problem of optimizing MEC power allocation, which is

$$\begin{aligned} (\mathcal{P}_3): \ & \min_{\mathbf{P}} \sum_{u\in\mathcal{U}} E_u \\ \text{s.t. (C1)}: \ & \mathrm{T_u} \le \mathrm{T_u^{max}}, \ \forall \mathrm{u} \in \mathcal{U}, \\ \text{(C2)}: \ & 0 \le p_u^{mec} \le p_u^{max}, \ \forall u \in \mathcal{U}, \end{aligned} \tag{16}$$

Since each cluster in the MEC utilizes orthogonal sub-carriers, there is no interference generated between clusters during power allocation. The power allocation method remains the same across clusters, so we will only focus on power allocation within a cluster. The objective of this paper is to reduce the energy consumption associated with task processing while ensuring user latency. Since multiple users share the same resources within a cluster, we propose a low-complexity power control strategy to enhance performance during demodulation.

(i) First, arrange users in descending order of channel gain, and for the last user, i.e., user K, let $T_K^{mec} = T_K^{max}$, and get p_K^{mec}.

(ii) After p_K^{mec} is given, according to formula (5), the transmit power of user $K-1$ can be obtained, i.e. p_{K-1}^{mec}.

(iii) When p_K^{mec} and p_{K-1}^{mec} are given, $p_{K-2}^{mec} \cdots p_1^{mec}$ can be obtained in turn by the same method.

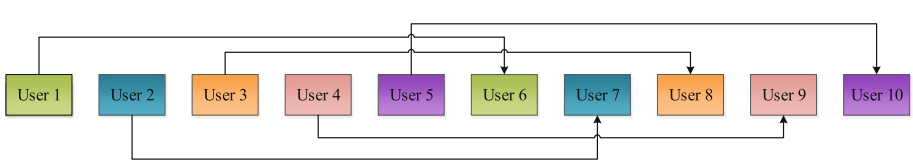

Fig. 2 Users pairing

For ease of understanding, we propose joint offloading decision, user clustering. The working flow chart of computing resource allocation and power allocation algorithm is shown in Fig. 3.

Simulation results

In this piece, we evaluate the advantages of the joint optimization algorithmic steps proposed in the paper. The parameter settings for the evaluation are displayed in Table 2.

This chapter will validate the advantages of our proposed algorithm. The proposed method is compared with each of the following two solutions. (i) Random offloading scheme: users randomly offload their work to MEC for processing. In addition, the user pairing, power and computing resources are divided in the way proposed in this paper. (ii) Random computing resource segmentation scheme: the computational resources are divided randomly, while the rest is in the same way as proposed in Table 2.

Figure 4 shows the trend of the total energy consumption of the system with the overall number of users. The total power consumption in this paper is the weighted sum of the transmission power consumption of user tasks and computing power consumption of server processing tasks. When the overall number of users increases, the total energy of the system is rising. It is observed from the figure that the aggregate system energy of each scheme rises with the increase of the aggregate number of users, which is consistent with the analysis result, but the aggregate system power consumption of the algorithm proposed in this chapter is the lowest. Due to the optimized allocation of transmission power and computational resources to the users, so the aggregate energy demand of the proposed scheme is smaller than the random power mechanism. Similarly, the scheme proposed in this text also optimizes user clustering, so the performance of the algorithm is the best as the number of users changes.

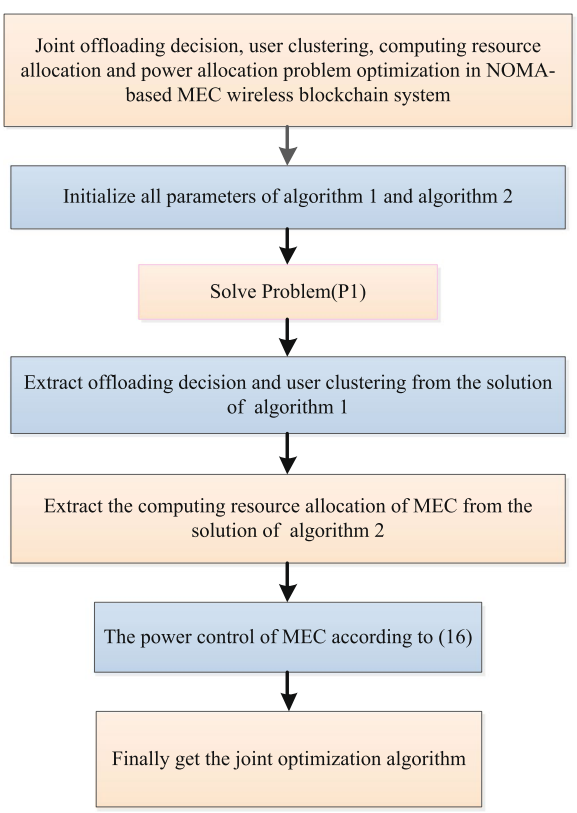

Fig. 3 The workflow of the joint optimization algorithm

Table 2 Simulation parameters

Parameter	Value
Number of users, N	50
Number of groups, M	4
Bandwidth of each group, B	400 Hz
Channel gain	$5 \sim 14$ randomly
Max send power of user u, p_u^{max}	0.2 W
Input data size, D_u	$1 \sim 20$ Kbit
Processing density, C_u	$100 \sim 500$
Local processing capability of user u, f_u^{max}	500 CPU cycles/s
Power of noise, σ^2	$10^{-7.4}$
Maximum tolerable latency, T_u^{max}	0.3 s

Figure 5 shows the impact of task offloading workloads C_u on the aggregate energy demand of the system. When C_u rises, the system energy requirements of the three algorithms increases for the following reasons. The first reason is that the higher C_u means fewer feasible tasks in local processing, and more users submit tasks to MEC server for processing, so more energy needs to be consumed. The second reason is that in MEC processing, the higher C_u means that task processing requires more resource blocks under a given tolerable delay constraint, thus consuming more energy, resulting in an increase in the total energy consumption of the system. It can be seen intuitively from the figure that the performance of the scheme proposed in this paper is the best.

In Fig. 6, the influence of maximum allowable delay on total energy requirements is plotted. When T_u^{max} increases, users do not need to allocate more resource blocks to meet the delay, that is, the number of resource pieces assigned to each user has become smaller. According to the objective function, the energy consumed by MEC server computing tasks will increase, thus resulting the energy demand of the system becomes larger. Since the benchmark algorithm does not optimize user clustering and resource allocation, the energy consumption

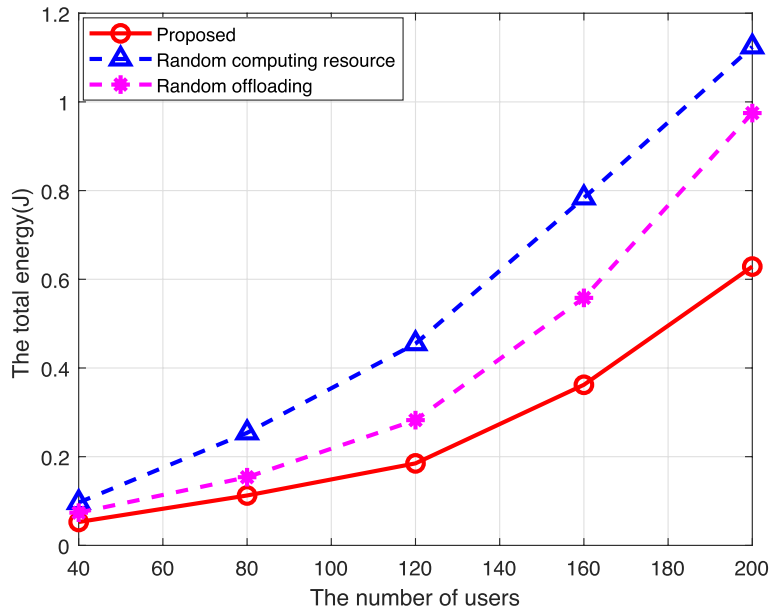

Fig. 4 Total energy vs. the number of users

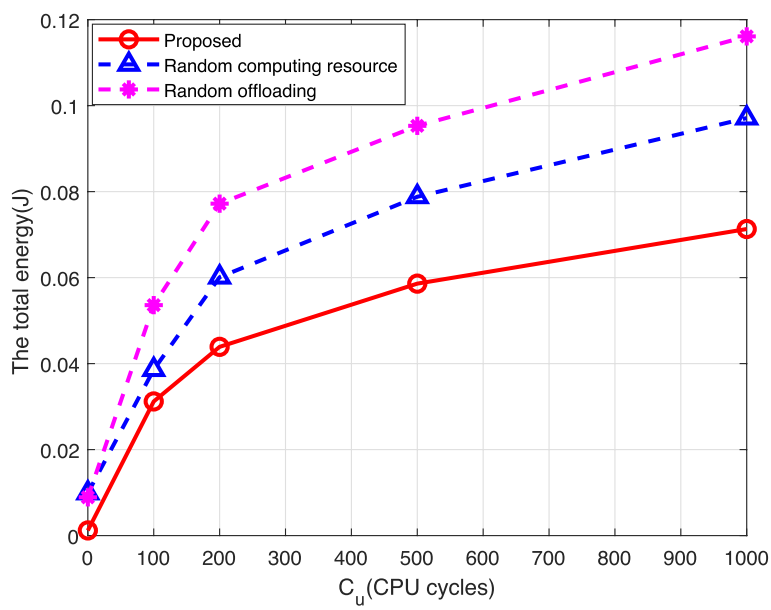

Fig. 5 Total energy vs. the computing intensity C_u

generated is greater than that of the proposed in the text. We can observe the idea in the paper from the picture is optimal in terms of energy consumption reduction.

Figure 7 shows how the application parameters of the input data size D_u affect the system energy consumption E_u. This graph is consistent with our understanding, that is to say, the energy demand will increase with the increase of input data D_u, and therefore the greater the total system energy consumption value. In addition, this method optimizes unloading decision and resource allocation at the same time, and its performance is always in the optimal state, followed by the random computing resource scheme, and the random offloading scheme has the worst performance.

Figure 8 shows the effect of local processing capability f_u^{max} on E_u, where the total system energy consumption

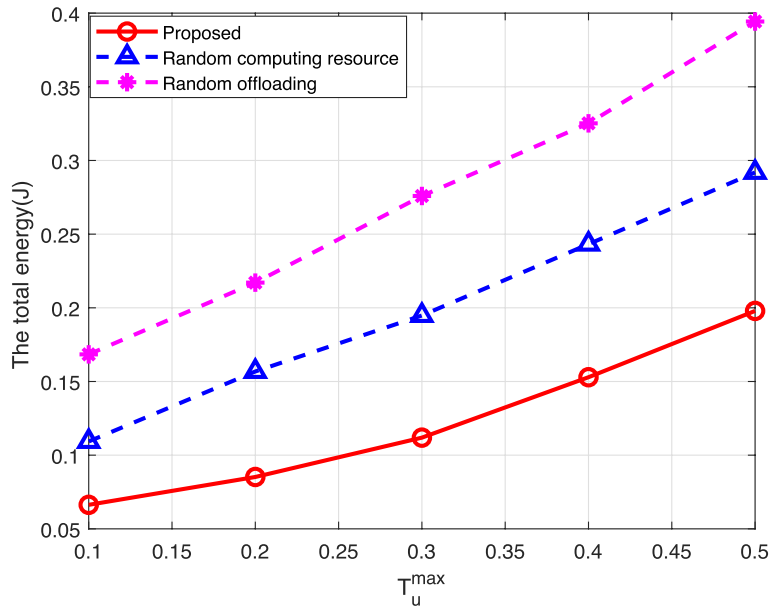

Fig. 6 Total energy vs. maximum tolerable latency T_u^{max}

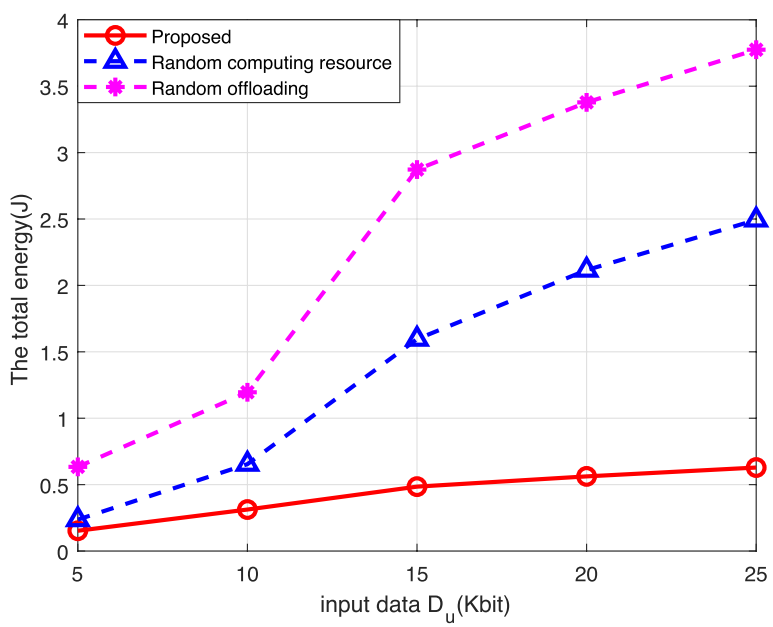

Fig. 7 Total energy vs. maximum tolerable latency D_u

increases rapidly as f_u^{max} increases for all algorithms. As f_u^{max} increases, when the user's local processing power is sufficient to support the responsibility of handling the task, it can handle most applications with good performance, so it does not need to transfer tasks to the MEC server. However, the resource joint optimization scheme figured out in this paper produces the lowest energy requirements. This is because the algorithm incorporates offloading decisions, MEC user clustering, and other related optimizations. The random offloading scheme does not optimize the assignment of tasks, so most of the assignments will be completed in the MEC server, and the resulting energy requirements is relatively large.

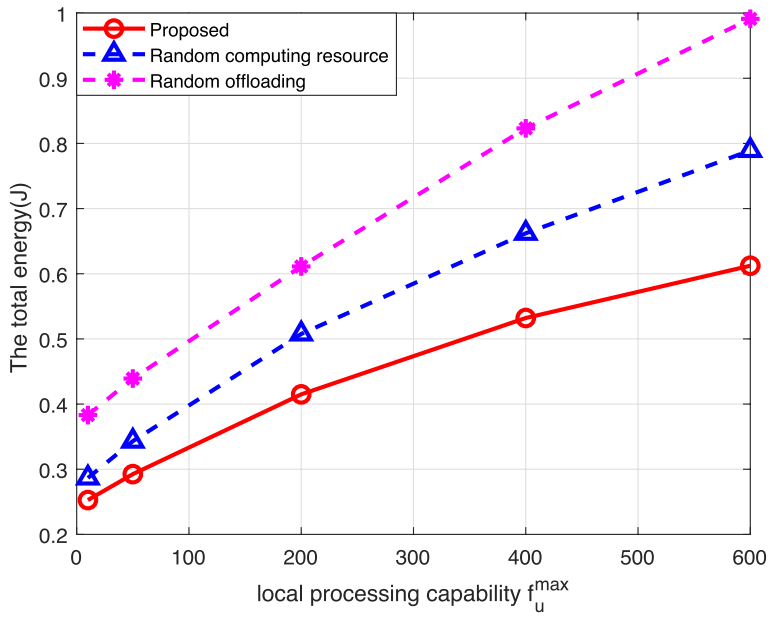

Fig. 8 Total energy vs. local processing capability f_u^{max}

Conclusions

In this paper, a low-energy optimization objective is proposed for a wireless blockchain system that combines NOMA and MEC. Consider a problem of minimizing system energy consumption, and through joint optimize offloading decisions, user clustering, computing resource and power assignment. Firstly, a low-complexity heuristic offloading decision and user clustering algorithm are proposed. Then, a computing resource block allocation algorithm is proposed based on the delay relationship, and finally the closed-form solution of the transmit power is obtained. Simulation experiments demonstrate the effectiveness of our proposed algorithm.

Acknowledgements
This work was supported in part by Basic Research Program of natural science in Shaanxi Province under grant No.2023-JC-YB-822, in part by Shaanxi Province's key Research and development project under grant No.2021-NY-211.

Authors' contributions
JianJie Ding, Lina Han and Jie Li wrote the main manuscript text and DaJun Zhang prepared releated works. All authors reviewed the manuscript.

Funding
This work was supported in part by Basic Research Program of natural science in Shaanxi Province under grant No.2023-JC-YB-822, in part by Shaanxi Province's key Research and development project under grant No.2021-NY-211.

Availability of data and materials
Not applicable.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare no competing interests.

Received: 30 January 2023 Accepted: 5 August 2023
Published online: 10 October 2023

References

1. Mao S, Liu L, Zhang N et al (2022) Reconfigurable Intelligent Surface-Assisted Secure Mobile Edge Computing Networks. IEEE Trans Veh Technol 71(6):6647–6660
2. Feng J, Liu L, Pei Q et al (2022) Min-Max Cost Optimization for Efficient Hierarchical Federated Learning in Wireless Edge Networks. IEEE Trans Parallel Distrib Syst 33(11):2687–2700
3. Li X, Li J, Liu Y, Ding Z, Nallanathan A (2020) Residual transceiver hardware impairments on cooperative noma networks. IEEE Trans Wirel Commun 19(1):680–695
4. Li X, Zhao M, Liu Y, Li L, Ding Z, Nallanathan A (2020) Secrecy analysis of ambient backscatter noma systems under i/q imbalance. IEEE Trans Veh Technol 69(10):12286–12290
5. Singh A, Aujla GS, Bali RS (2021) Intent-Based Network for Data Dissemination in Software-Defined Vehicular Edge Computing. IEEE Trans Intell Transp Syst 22(8):5310–5318
6. Guo H, Liu J, Zhang J (2018) Computation offloading for multi-access mobile edge computing in ultra-dense networks. IEEE Commun Mag 56(8):14–19
7. Wu Y, Ni K, Zhang C, Qian L, Tsang DHK (2018) NOMA assisted multi-access mobile edge computing: A joint optimization of computation offloading and time allocation. IEEE Trans Veh Technol 67(12):12244–12258
8. Ding Z, Xu J, Dobre OA, Poor HV (2019) Joint power and time allocation for NOMA-MEC offloading. IEEE Trans Veh Technol 68(6):6207–6211
9. Du J, Yu FR, Lu G, Wang J, Jiang J, Chu X (2020) Mec-assisted immersive vr video streaming over terahertz wireless networks: A deep reinforcement learning approach. IEEE Internet Things J 7(10):9517–9529

10. Feng J, Zhang W, Pei Q et al (2022) Heterogeneous Computation and Resource Allocation for Wireless Powered Federated Edge Learning Systems. IEEE Trans Commun 70(5):3220–3233
11. Du J, Cheng W, Lu G et al (2022) Resource Pricing and Allocation in MEC Enabled Blockchain Systems: An A3C Deep Reinforcement Learning Approach. IEEE Trans Netw Sci Eng 9(1):33–44
12. Zheng G, Xu C, Tang L (2020) Joint User Association and Resource Allocation for NOMA-Based MEC: A Matching-Coalition Approach. In: 2020 IEEE Wireless Communications and Networking Conference (WCNC). IEEE, Seoul
13. Liu B, Liu C, Peng M (2020) Resource Allocation for Energy-Efficient MEC in NOMA-Enabled Massive IoT Networks. IEEE J Sel Areas Commun 39(4):1015–1027
14. Fang F, Xu Y, Ding Z, Shen C, Peng M, Karagiannidis GK (2020) Optimal Resource Allocation for Delay Minimization in NOMA-MEC Networks. IEEE Trans Commun 68(12):7867–7881
15. Dai H-N, Zheng Z et al (2019) Blockchain for Internet of Things: A survey. IEEE Internet Things J 6(5):8076–8094
16. Kang J, Xiong Z et al (2019) Toward secure blockchain enabled Internet of Vehicles: Optimizing consensus management using reputation and contract theory. IEEE Trans Veh Technol 68(3):2906–2920
17. Hahn A, Singh R, et al (2017) Smart contract-based campus demonstration of decentralized transactive energy auctions. In: 2017 IEEE power & energy society innovative smart grid technologies conference (ISGT). IEEE, Washington, DC, p 1–5
18. Huang X, Zhang Y et al (2019) An optimal scheduling algorithm for hybrid EV charging scenario using consortium blockchains. Futur Gener Comput Syst 91:555–562
19. Zhang S, Lee J-H (2020) A group signature and authentication scheme for blockchain-based mobile-edge computing. IEEE Internet Things J 7(5):4557–4565
20. Yang T, Jiang Z et al (2020) Maritime search and rescue based on group mobile computing for unmanned aerial vehicles and unmanned surface vehicles. IEEE Trans Ind Inform 16(12):7700–7708
21. Xu Q, Su Z, Yang Q (2020) Blockchain-based trustworthy edge caching scheme for mobile cyber-physical system. IEEE Internet Things J 7(2):1098–1110
22. Liu M, Yu FR, Teng Y, Leung VCM, Song M (2019) Distributed resource allocation in blockchain-based video streaming systems with mobile edge computing. IEEE Trans Wirel Commun 18(1):695–708
23. Vera-Rivera A, Refaey A, Hossain E (2021) Blockchain-Based Collaborative Task Offloading in MEC: A Hyperledger Fabric Framework. In: IEEE International Conference on Communications Workshops (ICC Workshops). Montreal, IEEE, p 1–6
24. Zuo Y, Jin S, Zhang S (2021) Computation Offloading in Untrusted MEC-Aided Mobile Blockchain IoT Systems. IEEE Trans Wirel Commun 20(12):8333–8347
25. Feng J, Richard YuF, Pei Q, Chu X, Du J, Zhu L (2020) Cooperative Computation Offloading and Resource Allocation for Blockchain-Enabled Mobile-Edge Computing: A Deep Reinforcement Learning Approach. IEEE Internet Things J 7(7):6214–6228
26. Nguyen DC, Pathirana PN, Ding M, Seneviratne A (2020) Privacy-preserved task offloading in mobile blockchain with deep reinforcement learning. IEEE Trans Netw Serv Manag 17(4):2536–2549
27. Wu H, Wolter K, Jiao P, Deng Y, Zhao Y, Xu M (2021) EEDTO: An energy-efficient dynamic task offloading algorithm for blockchain-enabled IoT-edge-cloud orchestrated computing. IEEE Internet Things J 8(4):2163–2176
28. Su B, Liu H, Meng J (2021) Joint Time and Power Allocation for Wireless Powered NOMA-MEC Systems. In: 2021 13th International Conference on Communication Software and Networks (ICCSN). IEEE, Chongqing, p 24–28
29. Ding Z, Xu D, Schober R, Vincent Poor H (2021) Hybrid NOMA Offloading in Multi-User MEC Networks. IEEE Trans Wirel Commun. https://doi.org/10.1109/TWC.2021.3139932
30. Liu B, Liu C, Peng M (2021) Resource Allocation for Energy-Efficient MEC in NOMA-Enabled Massive IoT Networks. IEEE J Sel Areas Commun 39(4):1015–1027
31. Cheng Q, et al (2020) Efficient Resource Allocation for NOMA-MEC System in Ultra-Dense Network: A Mean Field Game Approach. In: 2020 IEEE International Conference on Communications Workshops (ICC Workshops). Dublin, IEEE, p 1–6
32. Zheng G, Xu C, Long H, Zhao X (2021) MEC in NOMA-HetNets: A Joint Task Offloading and Resource Allocation Approach. In: 2021 IEEE Wireless Communications and Networking Conference (WCNC). Nanjing, IEEE, p 1–6
33. He T, Wang D, Zhou F, Liang W, Tang X, Zhai D (2021) Delay-aware Offloading for Cooperative NOMA-based Near-and-far MEC Networks. In: 2021 IEEE/CIC International Conference on Communications in China (ICCC). IEEE, Xiamen, p. 978–983
34. Huynh LNT, Pham Q-V, Nguyen TDT, Hossain MD, Shin Y-R, Huh E-N (2021) Joint Computational Offloading and Data-Content Caching in NOMA-MEC Networks. IEEE Access 9:12943–12954
35. Wu W, Wang X, Zhou F, Wong K-K, Li C, Wang B (2021) Resource Allocation for Enhancing Offloading Security in NOMA-Enabled MEC Networks. IEEE Syst J 15(3):3789–3792
36. Xue J, An Y (2021) Joint Task Offloading and Resource Allocation for Multi-Task Multi-Server NOMA-MEC Networks. IEEE Access 9:16152–16163

Publisher's Note