Wireless Networks (2023) 29:2171–2187
https://doi.org/10.1007/s11276-023-03302-y

ORIGINAL PAPER

Low-latency AP handover protocol and heterogeneous resource scheduling in SDN-enabled edge computing

Chunlin Li[1,2,3] · Zhiqiang Yu[3,4] · Xinyong Li[5] · Libin Zhang[5] · Yong Zhang[3] · Youlong Luo[3]

Accepted: 22 February 2023 / Published online: 10 March 2023
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

As mobile devices are widely used and various applications emerge, users have higher demands on data rates and computing power. Software Defined Network (SDN) can configure and manage various devices in the network through a centralized control controller, making the network more flexible. In an SDN-enabled edge computing environment, dense multiple access devices make mobile devices handover frequently, and mobile devices handover between different access points becomes an inevitable problem. To address this problem, we propose an Access Point (AP) handover strategy based on the signal strength and traffic load. The scheme uses the global view and centralized control capability of the SDN controller to obtain, manage, and analyze information, then calculate the weights and compare them, and finally develop the handover policy. On the other hand, to improve system resource utilization and meet the performance demands of different applications, MEC systems need to allocate computing and communication resources appropriately to keep users' Quality-of-Service (QoS) experience. We propose a joint optimization strategy for computing and communication resources based on the Lagrange multiplier method. The policy calculates and analyzes the task execution latency and energy consumption of edge servers and local terminals, and develops an optimization scheme for sub-channel allocation and resource allocation. It aims to reduce latency and energy consumption as much as possible. The results of the experiments in this paper illustrate that the proposed AP handover scheme which is on the basis of received signal strength indicator (RSSI) and traffic load can effectively improve the task completion time and energy consumption performance. The proposed joint optimization strategy of computing and communication resources based on the Lagrange multiplier method can effectively improve energy consumption and delay performance.

Keywords SDN · MEC · Handover · Resource allocation

1 Introduction

As wireless technology becomes more widespread, mobile devices are widely used and the amount of data they generate shows exponential growth and diversification of data types, which puts higher demands on the computing power and requires a new way to process data that can compensate for the lack of hardware resources and at the same time achieve low latency, high reliability and low cost for the task. MEC technology has emerged for this situation [1–3]. It satisfied the demand for real-time data analysis, processing, and response of terminal devices, and realized the management of computing and storage resources [4, 5]. However, with the increasing scale of the network, the emergence of various applications, and the increasing number of users, issues such as user mobility still face

✉ Chunlin Li
chunlin74@tom.com

[1] Anhui Institute of Territorial Space Planning and Ecology, Anhui Jianzhu University, Hefei, People's Republic of China

[2] Key Laboratory of Agricultural Blockchain Application, Ministry of Agriculture and Rural Affairs, Beijing, People's Republic of China

[3] Department of Computer Science, Wuhan University of Technology, Wuhan 430063, People's Republic of China

[4] Robot Technology Used for Special Environment Key Laboratory of Sichuan Province, Mianyang, People's Republic of China

[5] State Key Laboratory of Smart Manufacturing for Special Vehicles and Transmission System, Baotou, People's Republic of China

many challenges, and the user demand for high data rates and high computing power needs to be further addressed [6–8]. The traditional network architecture with tightly coupled data forwarding and control logic has low flexibility and poor scalability, and to address the limitations of the traditional network architecture SDN emerged as a new network architecture [6, 9, 10]. It separates the forwarding function and control function of traditional networks and reduces the built-in functions of network devices by using a centralized control controller to configure and manage various devices in the network. The controller has an open programmable interface and will provide the user with a network-wide view and the system will become more flexible [11–13]. Users can access the status information of the underlying devices through the controller and can program policies such as traffic scheduling [14–16]. Endpoint handover in SDN-based network architecture can improve resource utilization and obtain better network service quality [17].

Although there are multiple access devices deployed at the network edge, it is difficult for users to find the most suitable access point from the wide range of network access points. In a dense multiple access device environment, frequent handover of user terminals will occur during mobility, which will lead to wasted channel resources and will reduce the user's quality of experience (QoE). The handover algorithm of mobile devices in multiple network access points will become very important. In addition, in MEC systems, the resources of the MEC system need to be properly allocated and optimized in order to make the computational tasks complete within their latency constraints and to allow more computational tasks to be processed by the MEC server. Resource allocation can improve the quality of service for users as well as enhance the computational performance of the system.

To avoid wastage of channel resources and ensure the quality of service for users, this paper proposes an AP handover algorithm based on the RSSI and traffic load. To address the resource allocation problem of the system, this paper presents a joint computational and communication resource allocation algorithm based on the Lagrangian multiplier method.

This paper has the following two main points.

(1) In this paper, we propose a handover solution based on RSSI and traffic load. The method uses the global view of the SDN controller and the centralized control capability of the SDN controller to obtain, manage and analyze information, then calculates the weights and compares them, and finally formulates a handover policy.

(2) In this paper, a joint communication and computing resource allocation method is proposed, and a model for joint optimization of sub-channels as well as computing resources is formulated with the objective of minimizing energy consumption and latency.

The remaining parts of this paper are structured like the following. Section 2 detailing the work involved in wireless AP handover and resource allocation. Section 3 models the wireless AP handover problem and a detailed description of wireless access point handover based on signal strength and traffic load. Section 4 establishes the resource allocation model as well as introduces the joint optimization algorithm of communication and computational resources based on the Lagrange multiplier method. Section 5 presents the comparative experimental results of the algorithm. Section 6 mainly summarizes the whole paper.

2 Related work

2.1 Handover

Suitable handover algorithms not only lower the consumption of channel resources, but also a better QoS for users, and there has been a lot of research on network AP handover in existing works[18–25]. Aldhaibani et al. [18] introduced an optimal handover (HO) algorithm in a new Software Defined Wireless Network (SDWN) architecture. The HO algorithm effectively improves QoE over dense wireless LANs. Wu et al. [19] put forward a handover skip solution for reference signal received power (RSRP) based in optical fidelity (LiFi) networks. The new method combines the RSRP value and its rate of change to determine the handover target. Gilani et al. [20] designed a new wireless LAN mobility management approach to handle load balancing and mobility management through Network Function Virtualization (NFV) technology and SDN and use NFV technology to reduce the problem of high handover latency. Chen et al. [21] presented a load balancing scheme for SD-WiFi networks that integrates the overload problem and the handover process. Ding et al. [22] proposed an authenticated handover mechanism under SDN domain (AHMMD). Kiran et al. [23] proposed an access point load balancing based handover scheme for SDN-WiFi system for efficient AP load based handover in overlapping wireless LAN cells. Zhang et al. [24] addressed the problem of frequent handover and handover failure in 5G ultra-dense networks, and proposed an unnecessary handover scheme in the light of understanding the user activity state of 5G networks and the new RRC inactivity state of 5G RRC layer. Gharsallah et al. [25] proposed a software-defined handover scheme to improve the handover in future 5G networks.

2.2 Resource allocation

Resource allocation can improve the quality of service for users as well as enhance the computational performance of the system, and the resource allocation problem has been studied in a number of existing works [26–33]. Ji et al. [26] presented an optimized framework based on deep learning to solve problem of resource allocation from wireless MEC. Xue et al. [27] studied the joint optimization of task offloading and resource allocation for multitasking and multi-server. They split the objective problem into two sub-problems, task allocation and resource allocation. Among them, resource allocation is subdivided into communication resource optimization and computational resource allocation. Wu et al. [28] proposed a pure DQN approach to solve the resource allocation and offloading policy problem with Q-learning (QL). Wu et al. [29] discussed the resource allocation problem in MEC networks and proposed a combined auction mechanism for combined resource allocation. Zhang et al. [30] presented a resource allocation and vehicle-to-everything offloading problem by introducing an SDN controller, giving computational resource allocation, transmission power control, offloading decisions and sub-channel allocation scheme. Aghdam et al. [31] present federated MEC optimization and wireless resource allocation for maximizing the energy efficiency of the system in a multi-layer NOMA HetNet. Du et al. [32] studied multi-user resource allocation and offloading decisions in a UAV-assisted MEC environment. Al-Razgan et al. [33] proposed a virtual machine resource algorithm for efficient task scheduling and resource allocation based on the feasibility of virtual machine availability in edge cloud servers.

3 Handover algorithm model based on RSSI and traffic load

3.1 Description of SDN-based seamless handover scenarios

The scenario description of the handover is shown in Fig. 1. We set the scenario to a single control domain in the SDN-based wireless access network, i.e., within the management domain of a single controller. The architecture mainly contains multiple wireless APs, SDN controllers, edge servers, base stations, and mobile terminal devices. The assumption is that the amount of access points is A and the amount of end devices is M. The set of end devices is denoted by $M = \{m_1, m_2, \ldots m_K\}$ and the set of APs is denoted by $A = \{a_1, a_2, \ldots, a_N\}$. Where it is specified that the terminal device can at most connect to one AP at the same time.

3.2 Parametric model affecting AP performance

The main idea of the load balancing terminal handover mechanism proposed in this chapter is as follows: using the global view of the SDN controller to collect the state information of all AP at the bottom and all mobile terminals associated with them; then integrating the collected state information according to the algorithm proposed in this chapter to calculate the weight value of each AP, determining the overall performance of each AP based on the weight value, and finally making the handover decision to select the AP with the best performance for the mobile terminal. Through this handover mechanism, the network load is balanced and the overall efficiency of the network is improved. In this chapter, signal strength, traffic load, and the number of connected terminals are selected as parameters.

3.2.1 Signal strength

Signal strength is one of the main parameters to judge the performance of the AP. Since the signal strength value may change from time to time due to the influence of geographic environment and other elements, in order to dilute the changes brought about by the influence of such external factors, this paper will do the following smoothing process on the received signal strength. The signal strength measured from the AP side is periodically obtained by the SDN controller and recorded to obtain the signal strength set S, which is expressed as formula (1).

$$S = \{s_1, s_2, \ldots, s_i, \ldots, s_R\} \tag{1}$$

S is the set of signal strength values acquired by the controller at every T_0 interval, and R is the number of signal strength values. Then, S is grouped in chronological order, and S is expressed as formula (2).

$$S = \{\{s_1, \ldots, s_N\}, \ldots, \{s_{iN+1}, \ldots, s_{JN+N}\}, \ldots, \{s_{(H-1)N+1}, \ldots, s_{(H-1)N+N}\}\} \tag{2}$$

where H is the amount of groups, N is the amount of signal strength in each group, H and N are taken as 15. Usually, the larger the value of signal strength, the relatively small impact of environmental interference, so each group in formula (2) will be sorted from the largest to the smallest rule, then take the first n numbers and sum them first, then take the average, and finally the set S' is obtained, the expression as in formula (3).

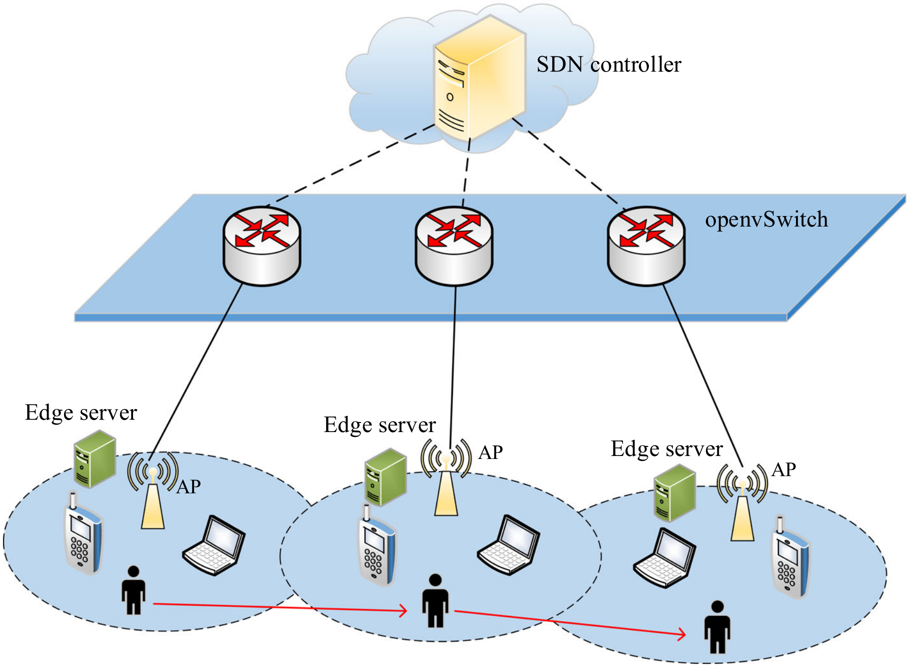


Fig. 1 handover scenes

$$S' = \{S'_1, S'_2, \ldots, S'_i, \ldots, S'_H\} \tag{3}$$

where S'_i is expressed as in Eq. (4).

$$S'_i = \frac{1}{n}\sum\nolimits_{n=1}^{n} s_n \tag{4}$$

3.2.2 Traffic load

Usually, the heavier the load, the poorer the AP's performance. To calculate the traffic load on each AP, this paper assumes that the terminal M on the connected AP receives IN_T bytes and sends OUT_T bytes in time period T. The unit traffic of terminal M in time period T is expressed as Eq. (5).

$$Q_M = \frac{IN_T + OUT_T}{T} \tag{5}$$

The overall traffic load on the AP is equal to the traffic information of all terminals connected to that AP, and the overall traffic load of the AP is shown in Eq. (6).

$$Q_{AP} = \sum_{i=0}^{k} Q_i \tag{6}$$

where k indicates the number of terminals connected to the AP.

3.2.3 AP weight calculation

Indicator weighting is an essential influence parameter, which can directly impact the evaluation results. The AP performance weight V is expressed as formula (7).

$$V = \alpha S' \times \frac{1}{\beta Q_{AP}} \tag{7}$$

Among them, the weight of the received signal strength and the bandwidth of AP occupied by the terminal in selecting the candidate handover user set can be adjusted by adjusting these two weight values to adjust the proportion of factors in constructing the candidate handover user set. SDN controller will maintain a weight table of each AP in the network.

3.3 Handover algorithm

This chapter proposes the terminal handover algorithm of the wireless access layer in the network. This chapter proposes the terminal handover algorithm for the wireless access layer in the network. When a terminal device is connected to a network first, the SDN controller will perform the authentication operation and then find the AP from the AP weight table that has the highest weight value accordingly to the terminal device to accomplish the connection of the device. In addition, due to the fluctuation of network services and the mobility of end devices, the weight table of APs needs to be constantly updated, and this part of the work will be done by the SDN controller. When the AP corresponding to a terminal device is detected to be unable to provide the best service, the AP with the highest weight value is selected for the terminal device to handover. The algorithm is as follows.

Algorithm 1: AP handover algorithm based on the RSSI and traffic load

```
Input: $S$ ,$k$ , $T$ , $IN_T$ , $OUT_T$
Output: APs that need to be switched by the terminal
1.  Within the time period T
2.  The signal strength received by the SDN controller is smoothed to obtain $S'$ .
3.  The traffic load $Q_{AP}$ of the AP is calculated from $IN_T$ and $OUT_T$ of each terminal .
4.  Calculated to obtain the weight table of AP
5.  For terminal $m_i$ , for AP weighting table
6.      if (the number of APs $m_i$ can connect to is equal to 0)
7.          end if
8.    else
9.          for each $j \in$ Connectable APs
10.             if ($V_{\mathrm{MAX}} <$ $V_{ij}$ ) // Compare weights
11.                 $V_{\mathrm{MAX}} = V_{ij}$ , Target AP becomes AP$_j$
12.             end if
13.           end for
14.     end if.
15. if (new update information is received before the switch is triggered, the weight table is
updated)
16.   goto 5
17. else
18.   if (Handover trigger)
19.       The terminal switches from the current AP to the target AP
20. end if
```

4 Heterogeneous resource allocation scheme

4.1 Resource allocation scenarios

The system architecture for resource allocation in this paper is shown in Fig. 2, which consists of an SDN controller, user terminals and multiple base stations configured with edge servers. Among them, the terminal devices are connected to the base stations through wireless networks, and the base stations can offload tasks to the remote cloud nodes via the Internet and accept the computation results from the remote cloud nodes' responses and return them to the terminal devices. Suppose there are N user terminals, each user requests multiple tasks at a time, and these tasks have the same data size and task complexity. Each of the user's tasks can be optionally offloaded to the edge server for execution or executed locally. A_i indicates the amount of tasks that end device i currently needs to perform. The amount of tasks performed locally by end device i is denoted by Q_i, and the task count offloaded to the edge server is represented by ${Q_i}^c$. The offloading decision constraint of endpoint i can be expressed as formula (8).

$$Q_i + Q_i^c = A_i(i \in \{1, 2, 3, \cdots, N\}) \tag{8}$$

It is assumed that each base station is provided with an edge server, each terminal device has a computing task to be completed, and the terminal devices in each base station operate in the same frequency band and interfere with each other. The number of base stations is denoted by M. The set of channels is denoted by $R = \{1, 2, 3, \cdots, r\}$. The offloading decision of a user is denoted by $D = [d_1, d_2, d_3, \ldots d_i]$. where d_i denotes the offloading decision of the ith terminal devices. Local execution of a task is denoted by $d_i = 0$ and offloading the task to the MEC server for execution is denoted by $d_i = 1$.

4.2 Communication model

Offloading compute to the edge for processing can reduce latency and energy consumption compared to local computing, but transferring task data consumes additional

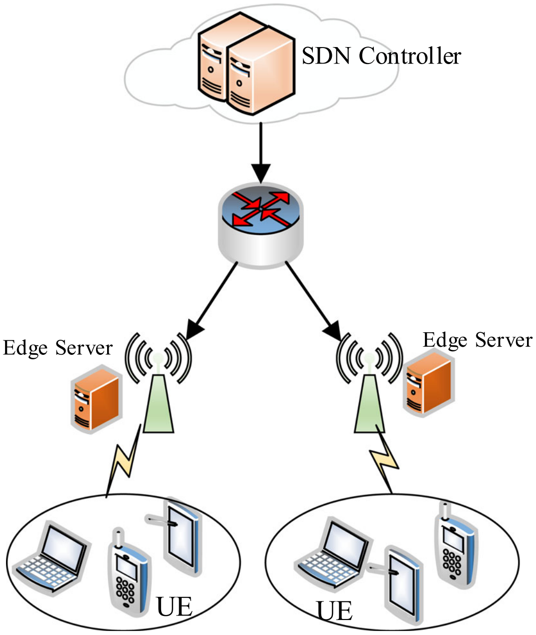


Fig. 2 system framework for resource allocation

latency and energy consumption. In addition, the amount of data sent by the device for request information is small compared to the task data offloaded to the edge, and the energy consumption and latency incurred for sending and receiving this information is usually not discussed in edge computing systems. According to the Shann-hartley theorem, the communication model from user i to base station m can be defined as formula (9).

$$R_{i,m} = b_{i,m}\omega \log_2\left(1 + \frac{q_i g_{i,m}}{\varphi^2 + \sum\limits_{j \in N\backslash\{i\}} q_j g_{j,m}}\right) \tag{9}$$

where ω denotes the channel bandwidth, $b_{i,m}$ denotes that if user i selects base station m for offloading, then $b_{i,m} = 1$, and vice versa $b_{i,m} = 0.q_i$ denotes the transmission power of user i, $g_{i,m}$ denotes the channel gain between user i and base station m, φ^2 denotes the Gaussian white noise power, and $q_j g_{j,m}$ denotes the interference caused by other users transmitting offloaded data to user i.

4.3 Computational model

The resource allocation strategy that is put forward in this paper aims at reducing the energy consumption of terminal devices and the latency of the system. Since the data level of the processed task is much lower than the data level when the task is generated, the energy consumption and time delay in the process of returning the computation results to the terminal are not considered in the calculation. The time delay of the system includes communication delay and computation delay. The communication latency includes the latency of transferring the task data from the terminal device to the base station, and the calculation latency includes the local execution latency and the MEC execution latency. The power consumption of the terminal includes transmission energy consumption and local computation energy consumption, which mainly includes the energy consumed by the user to compute the task itself and the energy required for uploading task information to the base station.

The computation task of user terminal i is defined as an array $U_i = \left(D_i, C_i, D_i^{out}\right)$, where D_i denotes the size of the input data for task computation,C_i denotes the number of CPU cycles required for single task execution,D_i^{out} denotes the size of the response data after task computation. In the following, we will discuss them both separately.

(1) Local execution

When the task is executed locally,$S_i = 0$, the maximum computing power limit of the mobile terminal and the computing power are represented by F_i^l and f_i^l; t_i^l and e_i^l denote the time and energy consumption required for the task to be executed locally, respectively. Thus, the time for the task can be expressed as formula (10).

$$t_i^l = \frac{C_i}{f_i^l} \tag{10}$$

Accordingly, the total energy consumption of the mobile terminal during the local execution of the task can be expressed as formula (11).

$$e_i^l = k\left(f_i^l\right)^2 C_i \tag{11}$$

where $k\left(f_i^l\right)^2$ is the energy consumption per CPU cycle of the mobile terminal for performing computing tasks. k is a constant, which is usually taken as $k = 10^{-26}$.

(2) Edge server execution

When offloading tasks to the edge server,$S_i = 1$, denoting the computing power of the edge server by f_i^c, has a maximum value of F_i^c;r_u denotes the data sending rate; $t_{i,m}^c$ and $e_{i,m}^c$ indicate the time and energy consumption of task offloading to the edge server for execution, respectively. Its time consumption includes both server execution time and transmission time as formula (12).

$$t_{i,m}^c = \frac{D_i}{r_u} + \frac{C_i}{f_i^c} \tag{12}$$

Similarly, the total energy consumption of a mobile terminal consists of data transmission energy consumption and mobile device idle energy consumption as formula (13).

$$e^c_{i,m} = p_i \frac{D_i}{r_u} + p_i^l \frac{C_i}{f_i^c} \tag{13}$$

where p_i denotes the mobile terminal i's transmitting power and p_i^I is the power when the device is idle.

Assuming that the tasks executed by the endpoints are with the same size, part of the batch of tasks can be handled by the edge server and locally at the same time. The terminal completes the task with the total energy consumption E as in Eq. (14). The terminal's total delay in completing the task T is as in Eq. (15).

$$E{=}\sum_{i=1}^{N}\sum_{m=1}^{M}\left(e_i^l + e^c_{i.m}\right) \tag{14}$$

$$T{=}\sum_{i=1}^{N}\sum_{m=1}^{M}\max\left(t_i^l, t^c_{i,m}\right) \tag{15}$$

4.4 Problem description

The objective of this chapter is to minimize the latency and energy consumption of the system-wide users in terms of computation time and energy consumption. Then the channel allocation Z and computational resource X allocation optimization problem for this user scenario can be expressed as Eq. (16).

$$\begin{aligned} &\text{C1 } \underset{Z,X}{min}\, W(Z,X) = E(Z,X) + \lambda T(Z,X) \\ &\text{C2 } \sum_{m=1}^{M} b_{i,m} \le 1;\ Q_i \ge 0, Q_i^c \ge 0 \\ &\text{C3 } z_i^r \in \{0,1\}, \forall r \in R, i \in \{1,2,\cdots,N\} \\ &\text{C4 } \sum_{i}^{N} z_i^r \le 1, \forall r \in R; \\ &\text{C5 } \sum_{r \in R} z_i^r \le 1, i \in \{1,2,\cdots,N\} \\ &\text{C6 } X \ge 0 \ \sum_{i}^{N} x_i \le X, x_i > 0; \end{aligned} \tag{16}$$

where λ denotes the weighting factor. The above C1-C6 equation carries some constraints. One more base station is selected for access. $z_i^r = 1$ denotes the assignment of channel r to user i. Otherwise $z_i^r = 0$, the MEC server assigns positive computing resources to the user. Only one user is allowed to access any channel and one channel is assigned to at most one user.

4.5 Channel and computational resource allocation algorithm

The assignment of user channels is first considered as a matching problem of users and channels, which is solved using matching theory. The preference relation for selection is formed by the preference functions of the user and the channel. Users and channels will be matched for preference selection to maximize transmission rate. The preference function of user i for channel r is given in Eq. (17).

$$V_i(r) = z_{ir} \tag{17}$$

Equation (17) displays the impact of the user's transmission rate for channel selection. The higher the transmission rate of user i on channel r, the smaller the offloading cost for that user. And each user tends to offload the computational task on the channel with higher transmission rate.

When channel r receives connection requests from different users, there are cases when different users request the same channel. In order to minimize the offloading cost, the channel r should select the user with the maximum transmission rate and the least interference from other users. Therefore, the preference function of channel r for user i is given in Eq. (18).

$$V_i(r) = \chi z_{ik} - \phi \xi_i^r \tag{18}$$

where χ is denoted as the unit rate gain factor of user i for transmission task on channel r, ϕ is the unit interference cost factor, and ξ_i^r is the total interference that user i receives from other offloaded users on channel r.

After the channel assignment is completed, the problem of computing resource allocation can be shown in Eq. (19).

$$\begin{aligned} &\min_{X} W(X) \\ &W(X) = \sum_{i}^{N} \frac{\gamma c_i}{x_i} \\ &\sum_{i}^{N} x_i \le X, x_i > 0 \end{aligned} \tag{19}$$

where x_i denotes the computational power allocated to user i for the server. γ denotes the computational latency weight for the user to perform the computer task. c_i denotes the CPU cycle required to perform the task.

The Hessian matrix of $W(X)$ consists of the elements $\frac{\partial^2 W}{\partial x_i^2} = \frac{2\gamma c_i}{x_i^3} > 0$ and $\frac{\partial W^2}{\partial x_i \partial x_n} = 0(i \ne n)$, and $W(X)$ is a semi-positive definite matrix. It is deduced that $W(X)$ is a convex function, and thus the problem of resource allocation is converted into a convex optimization problem. Furthermore, and based on the KKT condition that constrains

$x_i > 0$ to be relaxed, the Lagrangian function under the inequality constraint can be expressed as Eq. (20).

$$K(X,\alpha) = \sum_{i}^{N} \frac{\gamma c_i}{x_i} + \alpha\left(\sum_{i}^{N} x_i - X\right) \tag{20}$$

where $\alpha \geq 0$ is a Lagrangian multiplier.

Let $\partial K/\partial x_i|_{x_i^*} = 0$, then the optimal solution can be obtained as in Eq. (21).

$$x_i^* = \sqrt{\gamma c_i/\alpha} \tag{21}$$

Substituting Eq. (21) into (20), we can obtain the Lagrangian dual function of Eq. (19), expressed as Eq. (22).

$$\min_X K(X,\alpha) = 2\sqrt{\alpha}\sum_{i}^{N}\sqrt{\gamma c_i} - \alpha X \tag{22}$$

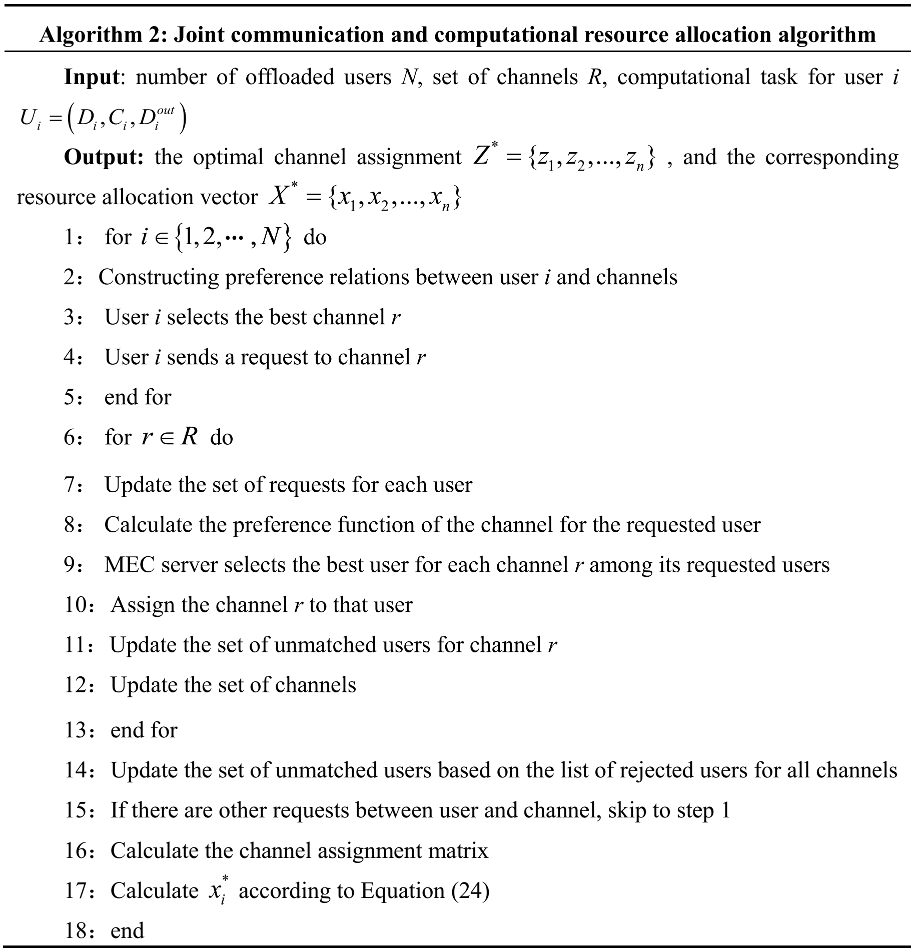
Next, we find the first-order partial derivative of α in Eq. (22) and make it equal to 0 to obtain Eq. (23).

$$\alpha^* = \left(\frac{\sum_{i}^{N}\sqrt{\gamma c_i}}{X}\right)^2 \tag{23}$$

Finally, the optimal computational resource x_i^* can be obtained, as in Eq. (24).

$$x_i^* = \frac{X\sqrt{\gamma c_i}}{\sum_{i}^{N}\sqrt{\gamma c_i}} \tag{24}$$

The core pseudo-code of the resource allocation algorithm is shown in Algorithm 2.



Algorithm 2: Joint communication and computational resource allocation algorithm

Input: number of offloaded users N, set of channels R, computational task for user i $U_i = \left(D_i, C_i, D_i^{out}\right)$

Output: the optimal channel assignment $Z^* = \{z_1, z_2, ..., z_n\}$, and the corresponding resource allocation vector $X^* = \{x_1, x_2, ..., x_n\}$

```
1： for $i \in \{1, 2, \cdots, N\}$ do
2：   Constructing preference relations between user $i$ and channels
3：   User $i$ selects the best channel $r$
4：   User $i$ sends a request to channel $r$
5： end for
6： for $r \in R$ do
7：   Update the set of requests for each user
8：   Calculate the preference function of the channel for the requested user
9：   MEC server selects the best user for each channel $r$ among its requested users
10：  Assign the channel $r$ to that user
11：  Update the set of unmatched users for channel $r$
12：  Update the set of channels
13： end for
14： Update the set of unmatched users based on the list of rejected users for all channels
15： If there are other requests between user and channel, skip to step 1
16： Calculate the channel assignment matrix
17： Calculate $x_i^*$ according to Equation (24)
18： end
```

5 Experiments

5.1 Experimental setup

(1) Experimental Setup for the proposed handover policy.

The experimental environment in this paper consists of multiple user devices, SDN controllers, and edge node servers. Table 1 shows the specific experimental parameter settings.

(2) Experimental Setup of proposed resource allocation strategy.

The experimental environment in this section consists of multiple end devices, SDN controllers, and edge node servers. The specific experimental parameters are set as shown in Table 2.

5.2 Analysis of experimental results on handover strategy

5.2.1 Comparison algorithms

The following handover scheme is compared with the RSSI and traffic load based handover scheme (RTF) presented for this paper.

(1) pre-scanning algorithm (PSA)

The literature [34] proposes a pre-scanning algorithm that starts channel scanning when the RSSI is less than the scanning threshold but not less than the handover threshold, and caches the dynamic information of the scanned APs so that when the RSSI is less than the handover threshold, the scanning phase can be directly crossed and the AP with the best signal is selected from the cache list for access and handover. RSSI-based handover algorithm is a more traditional terminal handover scheme in the handover algorithm, which generally takes the signal strength of each AP received by the terminal as the basis for handover.

(2) Adaptive load-balancing algorithm(ALBA)

The literature [35] designed an algorithm based on association control for adaptive load balancing in wireless software-defined networks. The algorithm derives an optimal scheme considering the amount of users and traffic load on each AP.

Table 1 Experimental key parameters

Parameters	Value
Number of APs	5
Local computing rate for users (GHz)	[0.5, 2.0]
Power in the transmit/receive state of the device (w)	[1, 4]
Power in the idle state of the device CPU (w)	[0.1, 0.6]

Table 2 Experimental key parameters

Parameters	Value
Number of subchannels	20
Subchannel bandwidth(MHz)	6
Local computation rate(GHz)	[0.5, 2.0]
Computation rate of edge server(GHz)	[1, 10]
Power in non-idle state of terminal device (w)	[1, 4]
Power in the idle state of the end device (w)	[0.1, 0.6]
The delay weight of calulation	[0.1, 0.9]
Gaussian channel noise (w)	2×10^{-13}

5.2.2 Algorithm performance metrics

To evaluate the RTF scheme for performance, handover delay, handover times, throughput, load balancing, and packet loss rate as algorithm performance evaluation metrics are selected.

(1) Time delay is the time experienced by the transmission of data from the sender side of a network to the receiver side, reflecting one of the important references of network service quality.

(2) The number of handover refers to a period of time the entire system terminal equipment to move the process occurs the number of handover.

(3) Load Balance shows the system's overall state at a point in time. We assume that all computing nodes have the same load situation at a certain moment. The calculation formula is shown in (25).

$$\eta = \frac{(load_i)^2}{(N \cdot load_i^2)} \tag{25}$$

where η is a quantitative scalar in the range of [0,1], the larger its value, the more balanced the load distribution among APs. N denotes the number of APs in the network, and $load_i$ denotes the workload of AP$_i$

(4) Throughput and packet loss rate are both important parameters to reflect the load situation.

5.2.3 Experimental results

Figures 3, 4, 5 and 6 show the effect of time variation on different performance parameters. As shown in Fig. 3, the variation of delay with time is compared for three different

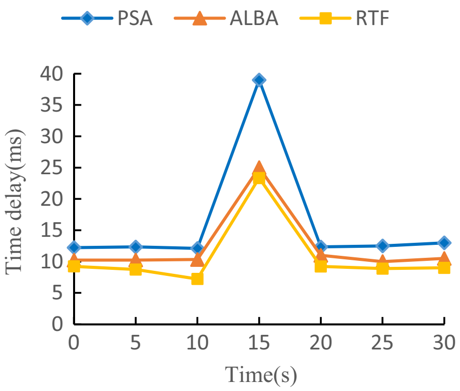


Fig. 3 Time delay versus time

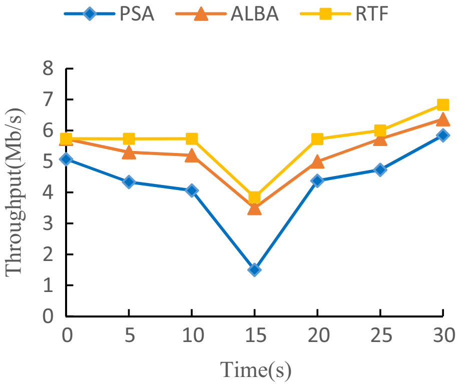


Fig. 4 Throughput over time

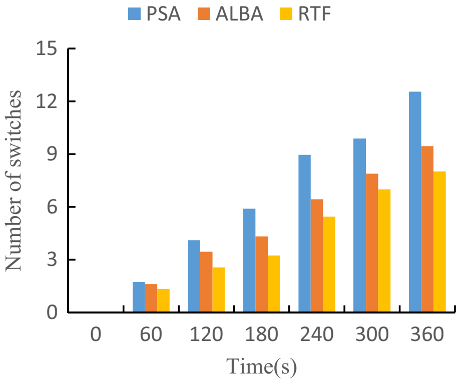


Fig. 5 Number of switches over time

handover algorithms. The variation of handover delay with time for a single user terminal, it can be clearly seen that the delay increases when a switchover occurs between 10 and 20 s at the terminal. At 5 s, compared to the PSA scheme, the latency of the RTF scheme is 29.2% lower, and compared to the ALBA scheme, the latency of the RTF scheme is 14.7% lower; at 15 s, compared to the PSA scheme, the latency of the RTF scheme is 40.2% lower, and compared to the ALBA scheme, the latency of the RTF scheme is 7.0% lower; when 25 s, compared to the PSA scheme, the latency of the RTF scheme is 28.7% lower, and compared to the ALBA scheme, the latency of RTF scheme is 11.2% lower. ALBA with this paper's algorithm uses an SDN controller, and the phase of authentication in handover is skipped, saving some time.

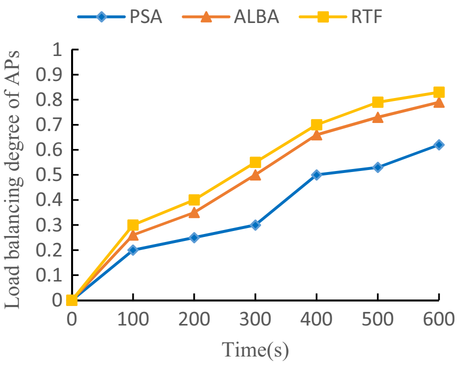


Fig. 6 Load balancing degree of APs over time

Figure 4 shows the variation of throughput with time. At 5 s, the throughput of the RTF scheme is 32.2% and 8.1% higher than that of the PSA and ALBA schemes, respectively. At 15 s, the throughput of the RTF scheme is 160.1% and 8.3% higher than that of the PSA and ALBA schemes, respectively. And when 25 s is reached, the throughput of the RTF scheme is 26.8% and 4.7% higher than that of the PSA and ALBA schemes, respectively. As shown in Fig. 4, a switchover occurs between 10 and 20 s, and the PSA scheme has a significant decrease in handover throughput when a switchover occurs. In contrast, the RTF scheme and the ALBA scheme switch more smoothly, and the drop in throughput varies less.

Figure 5 shows the average number of handover in the three schemes with time. As shown in the figure, at 120 s, the average number of handover is 22.3% lower for the RTF scheme compared to the PSA scheme, and 15.7% lower for the RTF scheme compared to the ALBA scheme; at 240 s, the average handover times of the RTF scheme are 37.4% and 15.6% lower than those of the PSA and ALBA schemes, respectively. At 300 s, the average handover times of the RTF scheme are 29.1% and 11.3% higher than those of the PSA and ALBA schemes, respectively. The PSA algorithm triggers the handover process frequently because of the signal strength which is subject to environmental interference factors and the variable size relationship. The ALBA scheme takes into account the number of users and traffic load. In contrast, the algorithm in this paper integrates the signal strength considering the AP traffic load and reduces many unnecessary handover times using SDN controller decisions.

Figure 6 illustrates the load balancing degree of APs in the three schemes over time. As shown in the figure, at

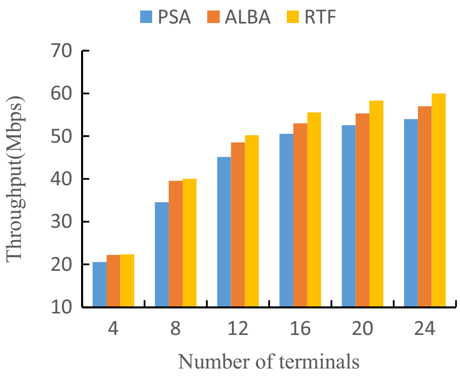


Fig. 7 Variation of throughput with the number of terminals

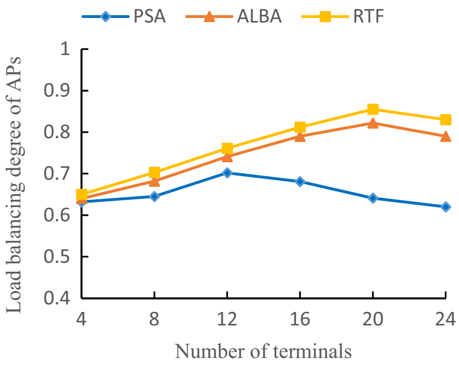


Fig. 8 Variation of load balancing degree with the number of terminals

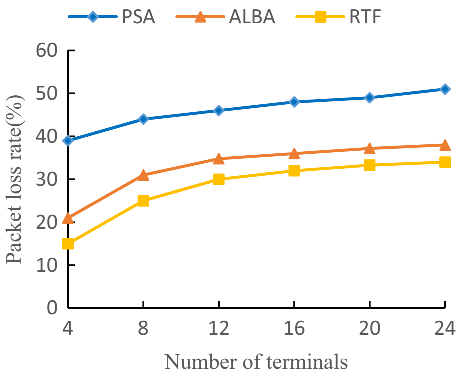


Fig. 9 Variation of loss packet rate with the number of terminals

200 s, the AP load balancing degree of the RTF scheme is 50.0% and 13.4% higher than that of the PSA and ALBA schemes, respectively. At 300 s, the AP load balancing degree of the RTF scheme is 78.8% and 10.5% higher than that of the PSA and ALBA schemes, respectively. When 500 s is reached, the AP load balancing degree of the RTF scheme is 49.1% and 8.2% higher than that of the PSA and The main handover factor of the PSA algorithm is the received signal strength, and as time goes by, the method of considering only the signal strength will lead to some APs having a higher load, while the resources of other APs are idle, resulting in a waste of resources. The RTF scheme and the ALBA scheme consider the load balancing situation, so it does not lead to load imbalance among APs.

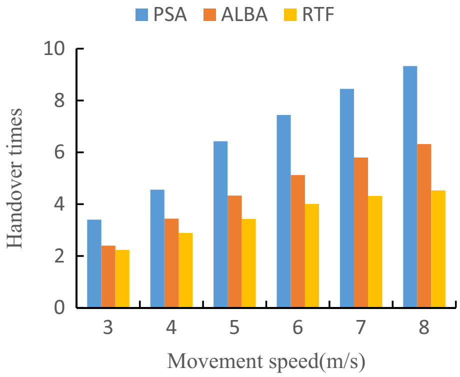


Fig. 10 Variation of handover times with motion speed

Figures 7, 8 and 9 illustrate the influence of the amount of user terminals on the results of the experiment. The experiment time is set to 1500 s.

Figure 7 presents the variation of system throughput with the amount of terminals during the handover of the three scenarios. As shown in the figure, the throughputs of all three handover schemes are gradually increasing with the amount of terminals. When the amount of terminals is 8, the RTF scheme has 16.0% and 1.4% higher system throughput than the PSA scheme and the ALBA scheme, respectively. At the amount of terminals is 20, the RTF scheme has 10.1% and 4.8% higher system throughput than the PSA and ALBA schemes, respectively. At the amount of 20 terminals, the RTF scheme has 12.5% and 5.5% higher system throughput than the PSA scheme and the ALBA scheme, respectively.

Figure 8 illustrates how the load balance of APs varies with the number of terminals during the handover process of the three schemes. When the number of terminals is 8, the load balancing degree of AP of RTF scheme is 9.1% and 3.1% higher than that of PSA scheme and ALBA scheme, respectively. When the number of terminals is 16, the load balancing degree of AP of RTF scheme is 19.3% and 5.9% higher than that of PSA scheme and ALBA scheme, respectively. When the amount of terminals is 20, the load balancing degree of AP of RTF scheme is 31.4% and 4.1% higher than that of PSA scheme and ALBA scheme, respectively. As shown in the figure, the load balancing degree of APs of all three handover schemes gradually increases and then decreases with the increase of the amount of terminals, and the PSA scheme selects the one with the largest signal strength for each handover, and the load balancing degree of APs increases with the increase of the number of terminals until the number of

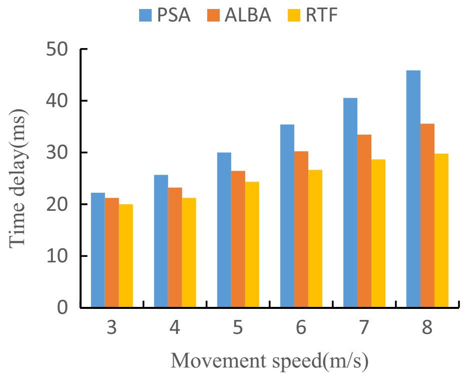


Fig. 11 Variation of time delay with motion speed

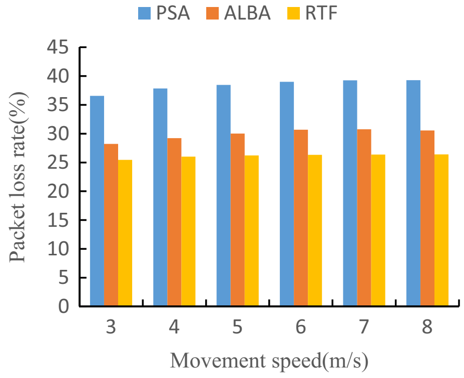


Fig. 12 The packet loss rate with the speed of movement

terminals reaches 12, and after that, the increase of the number of terminals causes some APs' resources to be idle, which leads to the unbalanced AP load. This leads to unbalanced load of APs. The ALBA scheme mainly considers the load balancing situation, and selects the AP with the lowest load in each switchover, and the load situation among APs will become more and more balanced as the amount of terminals increases. The proposed RTF algorithm in this paper takes into account the AP load situation in AP handover, so it does not lead to load imbalance among APs.

Figure 9 shows how the packet loss rate varies with the amount of terminals during the handover of the three schemes. As shown in the figure, the packet loss rate of all three handover schemes gradually increases as the amount of terminals increases, and the rate of increase is slow. When the number of terminals is 8, the packet loss rate of RTF scheme is reduced by 43.2% compared with PSA scheme, and 19.5% compared with ALBA scheme. When the number of terminals is 16, the packet loss rate of the RTF scheme is 33.3% reduced compared to the PSA scheme, and 11.1% reduced compared to the ALBA scheme. At the number of terminals of 20, the packet loss rate of the RTF scheme is reduced by 32.0% compared to the PSA scheme and by 10.5% compared to the ALBA scheme.

Figures 10, 11 and 12 illustrates the influence of user terminal movement speed on the results of the experiment. Set the experiment time to 1500 s and the number of user terminals to 15.

Figure 10 shows the change of handover times with user terminal motion speed in the handover process of the three schemes. At the user terminal motion speed of 3 m/s, the handover times of the RTF scheme are 34.4% and 7.0% lower than those of the PSA and ALBA schemes, respectively. At the user terminal motion speed of 5 m/s, the handover times of the RTF scheme are 45.7% and 21.1% higher than those of the PSA and ALBA schemes, respectively. At the user terminal motion speed of 8 m/s, the handover times of the RTF scheme are 50.4% and 28.1% higher than those of the PSA and ALBA schemes, respectively. The number of handover is 50.4% and 28.1% higher than that of PSA and ALBA schemes, respectively. As shown in the figure, the handover delay of all three handover schemes increases gradually with the increase of user terminal motion speed. This is because as the motion speed of the user terminal becomes larger, the time that the terminal stays in an AP decreases and the number of handover increases. It is obvious from the figure that the proposed algorithm reduces the amount of handover and avoids the ping-pong effect, and outperforms the PSA and ALBA schemes in terms of the number of handover.

Figure 11 illustrates the change in handover delay with the user terminal motion speed during the handover of the three schemes. At the user terminal motion speed of 3 m/s, the handover delay of the RTF scheme is 10.4% and 6.1% respectively less than the PSA solution and the ALBA solution. At the user terminal motion speed of 5 m/s, the handover delay of the RTF scheme is 18.2% and 5.8% respectively less than the PSA solution and the ALBA solution. At the user terminal motion speed of 8 m/s, the handover delay of the RTF scheme is 35.0% and 16.2% higher than that of the PSA and ALBA schemes, respectively. The handover delay of the RTF scheme is 35.0% and 16.2% higher than that of the PSA and ALBA schemes, respectively. As shown in the figure, the handover delay of all three handover schemes gradually increases with the increase of the user terminal motion speed. This is because as the motion speed of the user terminal becomes larger, the time that the terminal stays in the AP decreases, and then the handover delay becomes larger. It can be seen that the algorithm proposed in this paper outperforms the other two algorithms on the handover delay in the figure.

Figure 12 presents the change in packet loss rate with the speed of user terminal movement during the handover process of the three schemes. As the figure shows, when the

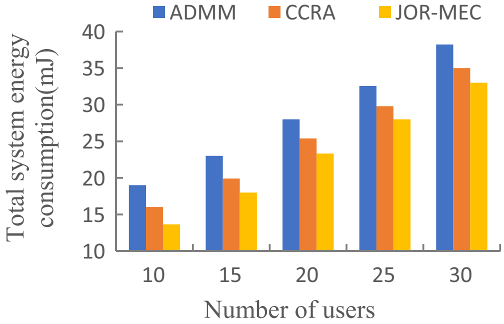


Fig. 13 Relationship between total system energy consumption and number of users

user terminal is moving at 3 m/s, the packet loss rate of the RTF scheme is 31.0% and 9.1% respectively less than the PSA solution and the ALBA solutions. At the user terminal speed of 5 m/s, the packet loss rate of the RTF scheme is 32.3% and 12.2% respectively less than the PSA solution and the ALBA solutions. At the user terminal motion speed of 8 m/s, the packet loss rate of the RTF scheme is 34.5% and 13.7% respectively less than the PSA solution and the ALBA solution. As shown in the figure, the packet loss rate of all three handover schemes gradually increases with the increase of the user terminal movement speed. the PSA scheme chooses the one which has the maximum signal strength for each handover, and there are more handover times and a lot of meaningful handover, which leads to a higher packet loss rate. From the figure, it is obvious that the proposed scheme in this paper outperforms another two schemes in terms of packet loss rate.

5.3 Analysis of experimental results on handover strategy

5.3.1 Comparison algorithm

We compare the following resource allocation solution with the Communication and Computational Resource Allocation Algorithm (CCRA) in this paper.

(1) alternating direction method of multipliers based algorithm (ADMM)

In the literature [36], an optimization problem for wireless cellular networks for MEC is proposed, and an algorithm based on alternating direction multipliers (ADMM) is proposed by transforming a convex problem into the original one.

(2) joint task assignment and resource allocation approach(JOR-MEC)

An approach based on the joint resource allocation and task assignment in a multi-user MEC architecture is proposed in the literature [37]. The method combines optimal offloading decisions with wireless resource allocation and aims to minimize the mobile terminal-side energy consumption under applied delay constraints.

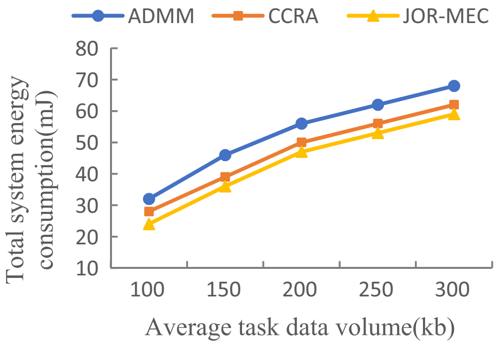


Fig. 14 Relationship between average task data volume and total system energy consumption

5.3.2 Experimental results

This experiment uses the control variables method and will simulate the user processing the computational task within 20 min.

Figure 13 illustrates the total system energy consumption as a function of the amount of terminal equipment. As the figure shows, the total system energy consumption of the CCRA scheme is 13.5% smaller than that of ADMM and 10.5% larger than that of JOR-MEC at the number of users of 15, and 8.4% smaller than that of ADMM and 6.1% larger than that of JOR-MEC at the number of users of 30. As can be seen from the figure, the total system energy consumption of each algorithm is increasing as the number of users increases.

Figure 14 illustrates the system total power consumption as a function of the average task data volume. As the figure shows, the total system energy consumption of the CCRA scheme is 15.3% smaller than that of ADMM and 8.3% larger than that of JOR-MEC at an average task data volume of 150 kb, and 8.8% smaller than that of ADMM and 5.1% larger than that of JOR-MEC at an average task data volume of 300 kb. From the figure, it is observed that the total system energy consumption of each algorithm increases as the average task data volume increases. The algorithm in this paper uses the centralized control function of SDN to achieve resource allocation and reduce energy consumption under the edge computing architecture, while the JOR-MEC scheme minimizes energy consumption under the application latency constraint.

Figure 15 presents the variation of the total task processing latency as a function of the user numbers. As the figure shows, the total task processing latency of the CCRA scheme is 30% smaller than ADMM and 41.6% smaller

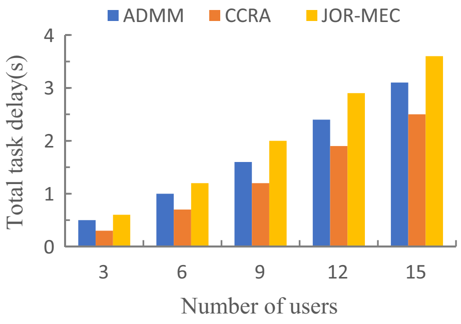


Fig. 15 Variation of total task latency with the number of users

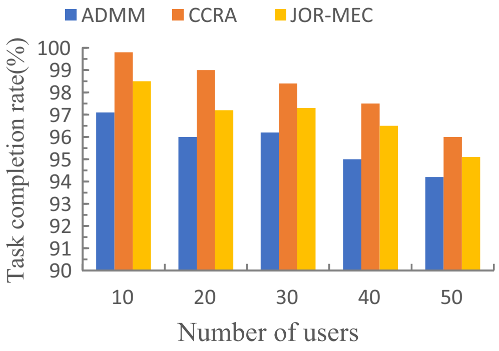


Fig. 16 Variation of task completion rate with the number of users

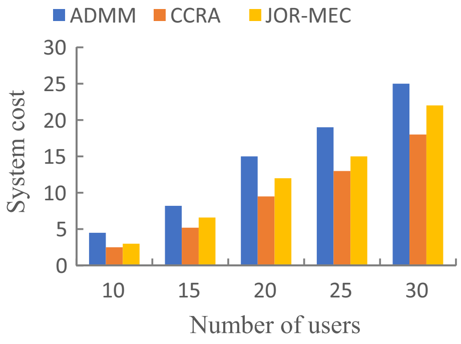


Fig. 17 Relationship between system cost and number of users

than JOR-MEC at the number of users of 6. At the number of users of 15, the total task processing latency of the CCRA scheme is 19.4% smaller than ADMM and 30.6% smaller than JOR-MEC. From the figure, we can see that as the amount of users becomes larger, the total task processing latency also enlarges. The increasing user number will result in an increase in the number of tasks, and task processing will require queuing in a queue, and the allocation of resources will result in longer processing latency and increased network transmission latency.

Figure 16 illustrates the variation of task completion rate as a function of the amount of users. At a user count of 10, the CCRA solution has a task completion rate 2.7% greater than ADMM and 1.3% greater than JOR-MEC. At a user count of 50, the task completion rate of the CCRA scheme is 1.8% greater than ADMM and 0.9% greater than JOR-MEC. The graph shows that as the user number increases, the task completion rate decreases. The amount of users increases, the amount of tasks increases, the demand for resources increases, and the number of tasks completed per unit of time decreases.

Figure 17 shows how the system cost (weighted sum of energy consumption and delay) varies with the number of users. As shown in the figure, the CCRA solution has a system cost 36.1% smaller than ADMM and 21.2% smaller than JOR-MEC at a user count of 15, and 28.0% smaller than ADMM and 18.2% smaller than JOR-MEC at a user count of 30. From the figure, with the increase of users, the amount of task computation will rise, and the system cost will also increase. Among all the algorithms, the proposed algorithm takes into account the effects of energy consumption and delay, and introduces SDN with the lowest cost of the system.

Figure 18 illustrates the variation of system cost with server processing power. As shown in the figure, the system cost of CCRA solution is 23.1% smaller than ADMM and 18.4% smaller than JOR-MEC when the server processing power is 2 GHz; and 10.8% smaller than ADMM and 5.4% smaller than JOR-MEC when the server processing power is 8 GHz. From the figure, the amount of task computation will increase with the increase of users and the system cost will also increase. The larger server processing capacity indicates that the server has sufficient resources for computing. The larger the edge server computational resources, the more computational resources are allocated to each offload task, and the system cost is naturally reduced. The system cost eventually converges when

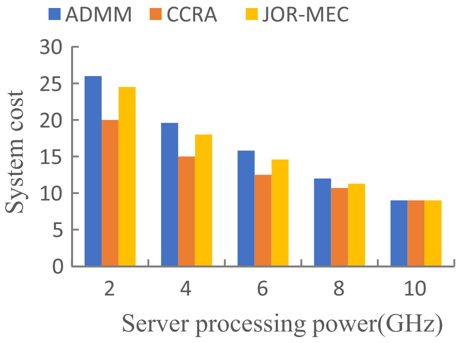


Fig. 18 Relationship between server processing power and system cost

the server processing power gets to a level where the tasks of all three algorithms can complete their tasks.

6 Conclusions

This paper proposes an AP handover algorithm based on the RSSI and traffic load. The method uses the global view of the SDN controller and the centralized control capability of the SDN controller to obtain and analyze information, then calculates the weights and compares them, and finally formulates a handover policy. In addition, this paper develops a model for joint optimization of sub-channels as well as computational resources with the objectives of minimizing the delay and energy consumption overhead of users, and presents a joint optimization algorithm for computing and communication resources.

There are still many shortcomings in this paper. (1) More factors can be considered when designing the AP handover algorithm to get better results. (2) When performing resource allocation, the joint optimization of computational offloading and resource allocation can be considered.

Acknowledgements The work was supported by by the National Natural Science Foundation ofChina (NSFC) under grants(No.62171330), Open Fund of Key Laboratory of Agricultural Blockchain Application, Ministry of Agriculture and Rural Affairs(No. 2022KLABA05) , Open Fund of Anhui Institute of Territorial Space Planning and Ecology (No.GTY2021101), Open Fund of Robot Technology Used for Special Environment Key Laboratory of Sichuan Province(No. 22Kftk05),.

Data availability No associated data.

References

1. Mao, Y., You, C., Zhang, J., et al. (2017). A survey on mobile edge computing: The communication perspective[J]. *IEEE Communications Surveys & Tutorials, 19*, 2322–2358.
2. Ab Ba, S. N., Yan, Z., Taherkordi, A., et al. (2017). Mobile edge computing: A survey[J]. *IEEE Internet of Things Journal, 5*, 450–465.
3. Zishu, L. I., Xie, R., Sun, L., et al. (2018). A survey of mobile edge computing[J]. *Telecommunications Science, 34*, 87–101.
4. Li, C., Liang, S. Y., Zhang, J., Wang, Q.-E., & Luo, Y. (2022). Blockchain-based data trading in edge-cloud computing environment[J]. *Information Processing and Management, 59*(1), 102786.
5. Chouhan, D., Gautam, N., Purohit, G., et al. (2021). A survey on virtualization techniques in mobile edge computing[J]. *WEENTECH Proceedings in Energy*. https://doi.org/10.32438/WPE.412021
6. Li, C., Zhang, Y., & Luo, Y. (2022). Intermediate data placement and cache replacement strategy under Spark platform[J]. *Journal of Parallel and Distributed Computing, 163*, 114–135.
7. Li, C., Zhang, Y., Gao, X., et al. (2022). Energy-latency tradeoffs for edge caching and dynamic service migration based on DQN in mobile edge computing[J]. *Journal of Parallel and Distributed Computing, 166*, 15–31.
8. Liu, J., Zhang, L., Li, C., Bai, J., Lv, H., & Lv, Z. (2022). Blockchain-based secure communication of intelligent transportation digital twins system. *IEEE Transactions on Intelligent Transportation Systems, 23*(11), 22630–22640.
9. Karakus, M., Durresi, A., et al. (2017). A survey: Control plane scalability issues and approaches in software-defined networking (SDN)[J]. *Computer Networks, 112*, 279–293.
10. Benzekki, K., Fergougui, A. E., & Elalaoui, A. E. (2016). Software-defined networking (SDN): A survey[J]. *Security & Communication Networks, 9*(18), 5803–5833.
11. Li, C., Cai, Q., & Youlong, L. (2022). Lowlatency edge cooperation caching based on base station cooperation in SDN based MEC. *Expert Systems with Applications, 191*, 116–252.
12. Li, C., Cai, Q., & Youlong, L. (2022). Optimal data placement strategy considering capacity limitation and load balancing in geographically distributed cloud[J]. *Future Generation Computer Systems, 127*, 100–111.
13. Liu, J., Zhang, L., Li, C., et al. (2022). Blockchain-based secure communication of intelligent transportation digital twins system[J]. *IEEE Transactions on Intelligent Transportation Systems*. https://doi.org/10.1109/TITS.2022.3183379
14. Li, C., Zhang, Y., & Luo, Y. (2023). DQN-enabled content caching and quantum ant colony-based computation offloading in MEC. *Applied Soft Computing, 133*, 109900.
15. Li, C., Zhang, Y., & Luo, Y. (2022). A federated learning-based edge caching approach for mobile edge computing-enabled intelligent connected vehicles. *IEEE Transactions on Intelligent Transportation Systems*. https://doi.org/10.1109/TITS.2022.3224395
16. Liu, J., Li, C., Bai, J., Luo, Y., Lv, H., & Lv, Z. (2023). Security in IoT-enabled digital twins of maritime transportation systems. *IEEE Transactions on Intelligent Transportation Systems, 24*(2), 2359–2367.
17. Oktian, Y. E., Lee, S. G., Lee, H. J., et al. (2017). Distributed SDN controller system: A survey on design choice[J]. *Computer Networks, 121*(5), 100–111.
18. Aldhaibani, O. A., Al-Jumaili, M. H., Raschella, A., et al. (2021). A centralized architecture for autonomic quality of experience oriented handover in dense networks[J]. *Computers & Electrical Engineering, 94*(2), 107352.
19. Wu, X., & Haas, H. (2019). Handover skipping for LiFi[J]. *IEEE Access*. https://doi.org/10.1109/ACCESS.2019.2903409
20. Gilani, S. M. M., Hong, T., Jin, W., et al. (2017). Mobility management in IEEE 802.11 WLAN using SDN/NFV technologies[J]. *Eurasip Journal on Wireless Communications & Networking, 2017*(1), 67.
21. Chen, Z., Manzoor, S., Gao, Y. et al. (2017). Achieving load balancing in high-density software defined WiFi networks[C]. In 2017 International conference on frontiers of information technology (FIT). IEEE Computer Society.
22. Ding, K., Wang, X., Zhang, G., et al. (2017). A flow-based authentication handover mechanism for multi-domain SDN mobility environment[J]. Wireless communication over ZigBee for automotive inclination measurement. *China Communications, 14*(9), 127–143.
23. Kiran, N., Yin, C., Akram, Z. (2017). AP load balance based handover in software defined WiFi systems[C]. In 2016 IEEE international conference on network infrastructure and digital content (IC-NIDC). IEEE.
24. Zhang, S. W., Wang, F. L., Yuan, C. J. (2019). Handover management strategy based on active state of 5G ultra-dense network users[J]. In Communications technology.
25. Gharsallah, A., Zarai, F., & Neji, M. (2019). SDN/NFV-based handover management approach for ultradense 5G mobile

networks[J]. *International Journal of Communication Systems, 32*, e3831.

26. Ji, L., Hui, G., Lv, T. et al. (2018). Deep reinforcement learning based computation offloading and resource allocation for MEC[C]. In 2018 IEEE wireless communications and networking conference (WCNC). IEEE.
27. Xue, J., & An, Y. (2021). Joint task offloading and resource allocation for multi-task multi-server NOMA-MEC networks[J]. *IEEE Access, 9*, 16152–16163.
28. Wu, Y. C., Dinh, T. Q., Fu, Y., et al. (2021). A hybrid DQN and optimization approach for strategy and resource allocation in MEC networks[J]. *IEEE Transactions on Wireless Communications, 20*, 4282–4295.
29. Wu, X., Jiang, W., Zhang, Y., et al. (2019). Online combinatorial based mechanism for MEC network resource allocation[J]. *International Journal of Communication Systems, 32*(7), e3928.1-e3928.16.
30. Zhang, H., Wang, Z., & Liu, K. (2020). V2X offloading and resource allocation in SDN-assisted MEC-based vehicular networks[J]. *China Communications, 17*(5), 266–283.
31. Aghdam, B., & Shaghaghi, K. R. (2021). Effective resource allocation and load balancing in hierarchical hetnets: Toward QoS-aware multi-access edge computing[J]. *The Computer Journal, 66*, 229–244.
32. Du, Y. (2021). Deep reinforcement learning for computation offloading and resource allocation in unmanned-aerial-vehicle assisted edge computing[J]. *Sensors, 21*, 6499.
33. Al-Razgan, M., Alfakih, T., & Hassan, M. M. (2021). A computational offloading method for edge server computing and resource allocation management[J]. *Journal of Mathematics, 2021*, 1–11.
34. Mustafa, N., Mahmood, W., Chaudhry, A. A., Ibrahim, C. M. (2005). Pre-scanning and dynamic caching for fast handoff at MAC layer in IEEE 802.11 wireless LANs. In IEEE International conference on mobile adhoc and sensor systems conference, pp. 8–122. https://doi.org/10.1109/MAHSS.2005.1542783.
35. Lin, C., Tsai, W., Tsai, M., Cai, Y. (2017). Adaptive load-balancing scheme through wireless SDN-based association control. In 2017 IEEE 31st international conference on advanced information networking and applications (AINA), pp. 546–553. https://doi.org/10.1109/AINA.2017.16.
36. Wang, C., Liang, C., Yuv, F. R. et al. (2017). Joint computation offloading, resource allocation and content caching in cellular networks with mobile edge computing[C]. In Icc IEEE international conference on communications. IEEE.
37. Dab, B., Aitsaadi, N., Langar, R. (2019). Joint optimization of offloading and resource allocation scheme for mobile edge computing[C]. In 2019 IEEE wireless communications and networking conference (WCNC). IEEE.

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Chunlin Li is a Professor of Computer Science in Wuhan University of Technology. She received the ME in Computer Science from Wuhan Transportation University in 2000, and PhD in ComputerSoftware and Theory from Huazhong University of Science and Technology in 2003. Her research interests include cloud computing and distributed computing.

Zhiqiang Yu received his bachelor's degree from Hubei University Of Automotive Technology in 2017. He is working toward master's degree in the School of Computer and Artificial Intelligence of Wuhan University of Technology. His research interests are edge computing and Internet of Vehicles.

Xinyong Li is a senior engineer of State Key Laboratory of Smart Manufacturing for Special Vehicles and Transmission System, dedicated to digital and intelligent manufacturing technology research.

Libin Zhang is an engineer of State Key Laboratory of Smart Manufacturing for Special Vehicles and Transmission System, dedicated to digital and intelligent manufacturing technology research.

Yong Zhang received his M.S. degree from Tiangong University in 2020. He is currently working toward the Ph.D. degree with the School of Computer Science and Technology, Wuhan University of Technology. His research interests include cloud computing and artificial intelligence.

Youlong Luo is a vice Professor of Management at Wuhan University of Technology. He received his M.S. in Telecommunication and System from Wuhan University of Technology in 2003 and his Ph.D. in Finance from Wuhan University of Technology in 2012. His research interests include cloud computing and electronic commerce.