IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received January 1, 2020, accepted January 16, 2020, date of publication February 6, 2020, date of current version February 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2972106

Intelligent Offloading Strategy Design for Relaying Mobile Edge Computing Networks

YINGHAO GUO[1], ZICHAO ZHAO[1], RUI ZHAO[1], SHIWEI LAI[1], ZOU DAN[2], JUNJUAN XIA[1], AND LISENG FAN[1]

[1]School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou 510006, China
[2]School of Information Engineering, East China Jiaotong University, Nanchang, 330013, China

Corresponding authors: Junjuan Xia (xiajunjuan@gzhu.edu.cn) and Liseng Fan (lsfan2019@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61971191, and in part by the Science and Technology Program of Guangzhou under Grant 201807010103.

ABSTRACT To support the application of IoT and smart city, high data-rate wireless transmission is required. To meet the demand of high data-rate, the techniques of multiple antennas and mobile edge computing (MEC) networks have been proposed in order to enhance the data transmission rate significantly. However, there still exist lots of challenges array signal processing assisted MEC networks. In this paper, we propose an intelligent framework of offloading strategy for MEC networks assisted by array signal processing, where one user with multiple antennas has some computational tasks. These tasks can be computed by the user itself which however has limited computational capability, or computed by the near-by computational access points (CAPs) which has a powerful computational capability at the cost of wireless transmission. We consider the system cost by jointly taking into account the computational price, the energy consumption and the latency. By minimizing the system cost, we propose an intelligent offloading strategy based on ant colony optimization (ACO) algorithm, where the ants randomly visit the CAPs in order to obtain the final results. To further enhance the MEC network performance, the array signal processing is utilized at the user, where either the maximum ratio transmission (MRT) or selection combining (SC) is used to assist the data transmission from the user to CAPs. Simulation results with MRT and SC are finally demonstrated to verify the effectiveness of the proposed ACO-based offloading strategy and array signal processing schemes.

INDEX TERMS Array signal processing, mobile edge computing, IoT, smart city.

I. INTRODUCTION

In recent years, the research and applications of smart city have attracted much attention, since it can help city planning, transportation planning, detecting unusual accidents, and so on [1]–[3]. To support the development of smart city, many new techniques have been proposed [4]–[6]. Among these techniques, the internet of things (IoT) technique has been proven to be a promising candidate [7], [8], which can intelligently acquire the city information, and process the obtained information [9], [10]. There are existing many researches on IoT and its applications, such as the urban environments detecting [11], [12]. Besides the above research, there have been some researches on the newly developed materials [13], [14] for IoT and smart city, which can be used in wireless networks for both transmission and improving the environments. In addition, there exist many researches on security to guarantee the wireless transmission in practice [15], [16], for the implementation of smart city.

Recently, there is an increasing trend that the users' demand has started switching from the traditional communication to computation [17]–[19]. In other words, many services in the wireless networks involve both the intensive computation and communication [20]–[22]. In some practical cases, the target is the computation result, while the communication is just a method. To support such kinds of new arising services, mobile edge computing (MEC) is proposed by the researchers in the filed of wireless communications. In MEC networks, the computational task can be offloaded to near-by nodes in the network which have a powerful computational capability, at the cost of wireless transmission. Hence, it is

The associate editor coordinating the review of this manuscript and approving it for publication was Liangtian Wan.

of vital importance to design an efficient offloading strategy to effectively offload the computational tasks into the near-by nodes, by taking into account the joint cost from the computation and wireless communication. In [23], a novel framework of offloading strategy was designed, where the offloading, the relay selection and the bandwidth allocation were jointly optimized to enhance the MEC network performance by reducing the cost of communication and computation substantially.

The application of intelligent algorithms such as deep learning and reinforcement learning based algorithms into wireless communication systems has attracted much attention, from the academia and industry [24]–[26]. The essence of these intelligent algorithms is the data-driven, instead of model-driven in the traditional works. In [25], [27], the deep learning was used to improve the detection performance of the maximum likelihood detector (MLD), by exploiting the local correlation characteristics through the deep convolutional neural network (DCNN). Besides the deep learning based works, the reinforcement learning algorithm can be also applied into the wireless communications. In [28], [29], the transmit power was intelligently optimized through the reinforcement algorithm, in order to tackle the smart attacker which can work in eavesdropping, interfering, spoofing and silent modes. In [30], [31], the authors used the antenna processing techniques and statistical channel state information of attacker, and effectively suppressed the smart attacker through the Q-learning algorithm.

In this paper, we propose an intelligent offloading strategy for MEC networks, where one user equipped with multiple antennas has some intensive computational tasks. These tasks can be computed by the user itself, or by the near-by computational access points (CAPs) through the help of wireless links. We consider the system cost as a linear combination of the computational price, the energy consumption and the latency. Based on this cost, we design the intelligent offloading strategy by the ant colony optimization (ACO) algorithm, in which the ants obtain a final offloading strategy through many times of random test. We further optimize the system performance to reduce the system cost by the array signal processing at the user. In particular, we employ two kinds of array signal processing at the user, i.e., maximum ratio transmission (MRT) and selection combining (SC), in order to assist the task transmission from the user to the CAPs. For the ACO based offloading strategy with the two array signal processing schemes, we finally present some simulation results in order to verify the proposed studies in this paper.

The organization of this paper is given as follows. After the introduction in this section, we will discuss the system model of MEC as well as the system model in Sec. II. Then, we introduce the two array signal processing schemes and the ACO based offloading strategy in Sec. III. Sec. IV will present the simulation results and conclusions are finally made in Sec. V.

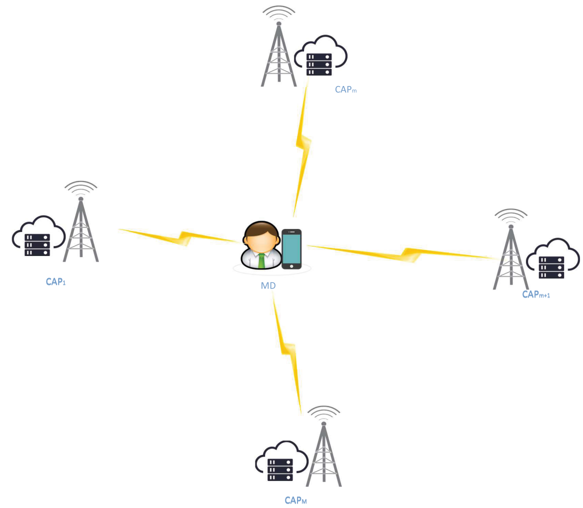


FIGURE 1. The whole system model in the Intelligent Offloading Strategy Design in Mobile Edge Computing Networks.

II. SYSTEM MODEL

Fig. 1 depicts the system model of task offloading in the MEC networks, where one user equipped with K antennas has some computational tasks assisted by the M near-by CAPs. In practice, the mobile user has limited computing capability in general, and hence it is difficult for the user to complete these tasks alone. Accordingly, the task is divided into N sub-tasks and these N sub-tasks can be offloaded to near-by CAPs. These CAPs have a powerful computational capability, and they can help the user finish the sub-tasks. In practice, the computational capability of the M CAPs is maybe different, and we use γ_m to denote the computational capability of the m-th CAP, where γ_m can be measured by the number of CPU cycles required for each bit in the sub-tasks. Moreover, the CAPs may have different charge prices for the user, and we use p_m to represent the computational price of the m-th CAP for computing 1M bits in the sub-tasks. In addition, we use l_n^{UL} and l_n^{DL} to denote the uplink and downlink bits for the n-th sub-task.

We use the symbol x_{nm} to indicate whether the n-th sub-task is offloaded or not. The value of x_{nm} is given by

$$x_{nm} = \begin{cases} 1, & \text{If the sub-task } n \text{ is offloaded to the CAP}_m \\ 0, & \text{If the sub-task } n \text{ is not offloaded to the CAP}_m. \end{cases} \tag{1}$$

From the formula (1), we obtain an offloading strategy matrix $\mathbf{X} = \{x_{nm} | 1 \leq n \leq N, 0 \leq m \leq M\}$. Note that each sub-task should be computed once only in order to save the computational resources, we have the following constraint accordingly,

$$\sum_{m=0}^{M} x_{nm} = 1. \tag{2}$$

If the n-th sub-task is computed by the user itself with $x_{nm} = 0$ for $m\forall[1, M]$, the computational latency and energy consumption are given by

$$t_{n0}^{comp} = \frac{l_n^{UL}}{\gamma_0}, \tag{3}$$

$$e_{n0}^{comp} = P^{comp}\frac{l_n^{UL}}{\gamma_0}, \tag{4}$$

where γ_0 is the computational capability of the user's CPU, and P^{comp} is the user's computational power. Notations t_{n0}^{comp} and e_{n0}^{comp} represent the locally computational latency and local energy consumption, respectively.

On the other hand, if the n-th sub-task is computed by the CAP$_m$ with $x_{nm} = 0$ for $m \in [1, M]$, the uplink and downlink transmission time are given by

$$t_{nm}^{UL} = \frac{l_n^{UL}}{C_m^{UL}}, \tag{5}$$

$$t_{nm}^{DL} = \frac{l_n^{DL}}{C_m^{DL}}, \tag{6}$$

where t_{nm}^{UL} and t_{nm}^{DL} are the transmission time of the uplink and downlink, respectively. Notations C_m^{UL} and C_m^{DL} denote the transmission data rate of the uplink and downlink, respectively, which will be detailed in the next section. From t_{nm}^{UL} and t_{nm}^{DL}, the transmission energy are given by

$$e_{nm}^{UL} = P_{trans}^{UL}\frac{l_n^{UL}}{C_m^{UL}}, \tag{7}$$

$$e_{nm}^{DL} = P_{trans}^{DL}\frac{l_n^{DL}}{C_m^{DL}}, \tag{8}$$

where P_{trans}^{UL} and P_{trans}^{DL} are the transmit power of the uplink and downlink, respectively.

In addition, the computational time for the n-th sub-task executed at the CAP$_m$ is given by,

$$t_{nm}^{comp} = \frac{l_n^{UL}}{\gamma_m}, \tag{9}$$

where the γ_m is the computational capability of the CAP$_m$. Since CAPs are directly connected to the power supply in general, the energy consumption at the CAPs is ignored in this paper. The charge of computing the user's n-th sub-task by CAP$_m$ is given by

$$\Lambda_{nm} = p_m l_n^{UL}. \tag{10}$$

From the above description, we can obtain the latency which CAP$_m$ requires to calculate the assigned sub-tasks, given by

$$T_m(X) = \begin{cases} \sum\limits_{n\in N} x_{nm}(t_{n0}^{comp}), & m = 0 \\ \sum\limits_{n\in N} x_{nm}(t_{nm}^{UL} + t_{nm}^{DL} + t_{nm}^{comp}), & m > 0. \end{cases} \tag{11}$$

In this equation, $m = 0$ indicates that the latency of user's local CPU to complete the assigned sub-tasks is given by T_0; while $m > 0$ represents that the sub-tasks are offloaded to the near-by CAPs, where the associated transmission latency and computing latency are given by T_m. In this work, we consider that the sub-tasks are transmitted from the user to the CAPs in parallel, and the CAPs can work in parallel, the latency of completing all the sub-tasks is the maximum latency that M CAPs complete their individual sub-tasks,

$$T_{total} = \max_{m\in\mathcal{M}} T_m(X). \tag{12}$$

The system total energy consumption is defined as the sum of the transmission energy and the computational energy. Accordingly, the system total energy consumption is given by

$$E_{total} = \sum_{m=0}^{M}\sum_{n=1}^{N} x_{nm}(e_{nm}^{comp} + e_{nm}^{UL} + e_{nm}^{DL}). \tag{13}$$

The system total charge from the CAPs is given by

$$\Lambda_{total} = \sum_{m=1}^{M}\sum_{n=1}^{N}(x_{nm}\Lambda_{nm}). \tag{14}$$

From the latency T_{total}, energy consumption E_{total} and the charge Λ_{total}, we can obtain the three performance metrics to measure the system performance of the MEC networks. In the following section, we will describe how to formulate the optimization problem and how to solve the optimization problem for the considered MEC networks.

III. ARRAY SIGNAL PROCESSING AND ACO-BASED OFFLOADING STRATEGY

In this section, we could first present several array signal processing schemes, which affect the upload and download data rates of wireless transmission. Then, we will present a joint performance metric to measure the system performance of the MEC networks, based on the latency, energy consumption, and the price. After that, we will describe the ACO-based offloading strategy design for the considered MEC networks.

A. ARRAY SIGNAL PROCESSING

In this work, multiple antennas are equipped at the user, which can be utilized through away signal processing to exploit the gain from multiple antennas. A simple way to exploit the multiple antennas is the random antenna selection scheme, which is also equivalent to the usage of only one antenna at the user. Accordingly, the upload and download data rates with the random antenna selection(RAS) are given by:

$$C_m^{UL} = B\log_2(1 + \frac{P_{trans}^{UL}|h_1|^2}{\sigma^2}). \tag{15}$$

$$C_m^{DL} = B\log_2(1 + \frac{P_{trans}^{DL}|h_1|^2}{\sigma^2}). \tag{16}$$

where B is the wireless bandwidth between the user and CAPS, and $h \sim \mathcal{CN}(0, \zeta)$ denotes the channel parameter between the 1-st antenna of the user and the CAPs. Notations σ^2 is the noise power of the additive white Gaussian noise (AWGN) [32]–[35], where the effect of noise on the communication systems can be found in the literature such as the works [36]–[39].

Besides this array signal processing method, there exist some other schemes to exploit the multiple antennas at the user. In particular, selection combining (SC) scheme can be used to choose the best antenna at the user [40]–[43], which can maximize the upload and download data rates of wireless transmission when some antennas are used. Accordingly, the upload and download data rates with SC are given by:

$$C_m^{UL} = B\log_2(1+\frac{P_{trans}^{UL}\max\limits_{1\le k\le K}|h_k|^2}{\sigma^2}). \tag{17}$$

$$C_m^{DL} = B\log_2(1+\frac{P_{trans}^{DL}\max\limits_{1\le k\le K}|h_k|^2}{\sigma^2}). \tag{18}$$

In addition to the RAS and SC, MRT can be used to exploit all the benefits from multiple antennas, at the cost of using multiple RF chains, which increase the system implementation complexity. Accordingly, the upload and download data rates with MRT are given by:

$$C_m^{UL} = B\log_2(1+\frac{P_{trans}^{UL}\sum_{k=1}^{K}|h_k|^2}{\sigma^2}). \tag{19}$$

$$C_m^{DL} = B\log_2(1+\frac{P_{trans}^{DL}\sum_{k=1}^{K}|h_k|^2}{\sigma^2}). \tag{20}$$

B. JOINT PERFORMANCE METRICS

The MEC networks include the latency, energy consumption and the price, which present three-dimensional performance metrics. We can measure the system performance through the three-dimensional perspectives, which how cause much difficulty in system optimization. To simplify the measurement on the system performance and the associate of optimization, we consider a joint performance metric based on the latency, energy consumption and the price, given by

$$\Phi = \lambda_1 E_{total} + \lambda_2 T_{total} + (1-\lambda_1-\lambda_2)\Lambda_{total} \tag{21}$$

with

$$0 \le \lambda_1 \le 1, \tag{22}$$

$$0 \le \lambda_2 \le 1, \tag{23}$$

$$0 \le \lambda_1+\lambda_2 \le 1, \tag{24}$$

where λ_1, λ_2 and $(1-\lambda_1-\lambda_2)$ denote the weight factors of the latency, energy consumption and the price in the overall cost, respectively. In particular, when λ_1 becomes larger, the energy consumption plays a more important role in the system whole cost; when λ_2 increases, the latency becomes more important in the system cost; when the value of $(1-\lambda_1-\lambda_2)$ becomes larger, the price presents a more important role in the system cost. Moreover, it should be noted that the joint performance metric Φ can reflect the three dimensional performance metrics to some extent. Specifically, when $\lambda_1 = 1$ holds, the joint cost Φ degenerates into the energy consumption; when $\lambda_2 = 1$ holds, the joint cost Φ degenerates into the latency; when $\lambda_1 = \lambda_2 = 0$ holds, the joint cost Φ degenerates into the price.

C. OFFLOADING STRATEGY OPTIMIZATION

After the array signal processing at the user, we obtain the optimized uplink and downlink data rates for the wireless transmission between the user and CAPs. Based on the joint cost Φ, we will describe hoe two devise the offloading strategy, in order to minimize the system cost. In this work, we employ the intelligent ACo scheme to obtain the offloading matrix $\mathbf{X}$. In general, the ACO is a bionic optimization algorithm that simulates the foraging behavior of ants, and it is also a type of heuristic algorithm. It was firstly used to solve the traveling salesman problem (TSP), which is a classic NP problem. And the goal of TSP is to find the shortest traversal route for a series of cities. To apply the ACO to solve the problem of task assignment in MEC networks, we change the optimization goal from obtaining the shortest path to achieving the minimal cost. In addition, there is a concept of pheromone in ACO, which plays a leading role in the optimization. After each ant matches a task with a given CAP, it will leave a pheromone on this choice. Then the ant colony will make a corresponding decision based on the pheromone of all choices, and finally select the best allocation strategy for the task offloading in MEC networks.

Specifically, to implement the ACO scheme, we firstly define a pheromone concentration matrix γ, which is used to record the pheromone concentration on the ant matching path. Then, we let each ant choose a sub-task by random, and each sub-task can be selected once only. We use a matrix $\mathbf{O}$ to record whether the sub-tasks have been selected or not. And each ant matches a CAP to it after selecting a task. Ants stop crawling after all tasks have matched all CAPs. This process of matching the CAPs is a process of probability selection. And the probability selection here is a roulette probability selection, so that each selection may select an event with a low probability to ensure the randomness of selection. We define the selection probability as P, which can be expressed as:

$$P_{nm} = \frac{\gamma^{\alpha}(n, m_j)\Phi^{\beta}(n, m_j)}{\sum_{m=0}^{M}\gamma^{\alpha}(n, m)\Phi^{\beta}(n, m)} \tag{25}$$

where m_i indicates the currently selected CAP, $\alpha \in [0, 5]$ is the information heuristic factor and $\beta \in [0, 5]$ is the except heuristic factor. The larger the value α, the greater the possibility that the ant chooses the path previously traveled. Accordingly, the randomness of the search path decreases. When the ant colony search range becomes smaller, it is easier for the search to fall into a local optimum. With a larger value β, it is easier for the ant colony to choose a local short path. At this time, although the convergence speed of the ACO algorithm is fast, the randomness is however decreased, which may lead to a local optimum.

We call all ants crawling as one time of iteration. We use i as to denote iteration index, which is in the range of $[1, I]$ value for each iteration and I for the total number of iterations. The corresponding pheromone will be left on the distribution path after each iteration, and then we start to

Algorithm 1 ACO Algorithm for Optimization

for each $i \in [1, I]$ **do**
  for each ant in ants **do**
    for each $n \in [1, N]$ **do**
      Randomly select a task among N ones
      if $\mathbf{O}[n]$ **then**
        Select n as the next task and set $\mathbf{O}[n]$ to false.
      end if
      for each $m \in [1, M]$ **do**
        Compute the cost of each allocation strategy by 21
        Compute the possibility of each allocation strategy by 25
      end for
      Roulette selects an allocation strategy
      Set the $\mathbf{X}[n, m]$ to 1
    end for
    Obtain a compete allocation matrix $\mathbf{X}$
    Compute the system cost with $\mathbf{X}$ by 21
  end for
  Select the best ant in this iteration
  Update the pheromone of each allocation method by the 26
end for

update the pheromone concentration matrix γ. The update formula can be expressed as:

$$\gamma_{nm}^{i} = (1-\rho)\gamma_{nm}^{i-1} + \Delta\gamma_{nm}^{i} \tag{26}$$

where $\rho \in [0, 1]$ is an information volatility factor which is used to affect the concentration change of pheromone, and $\Delta\gamma$ represents the pheromone left by the ant after the current i-th iteration. When ρ is very small, there is a lot of pheromone remaining on an allocation path, which may cause invalid allocation paths to continue being searched and affect the convergence rate of the algorithm. On the contrary, when ρ is very large, there is little guarantee that the valid path will continue to be searched although invalid paths can be excluded from searching. Moreover, the value of ρ will affect the search result of the optimal value. In addition, the right side of this equation indicates that the information concentrations on the distribution path at the $(i-1)$-th time. The left side of this formula indicates that the distribution concentrations on the distribution path after the current i-th iteration.

After each iteration, we can find an optimal ant whose total cost on the distribution path is the minimum. After going through times of iterations, the allocation matrix of the optimal ant is the final result that we want to obtain. In this way, we have completed the ACO-based offloading strategy for the MEC networks. In the table of Algorithm, we have summarized the MEC-based offloading strategy for the MEC networks.

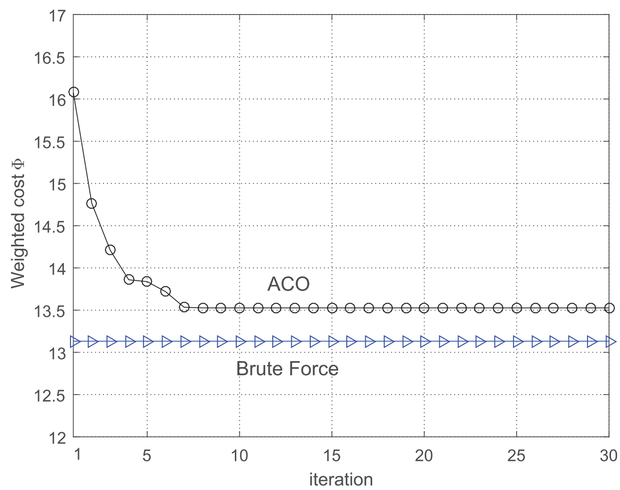


FIGURE 2. Performance of the ACO algorithm with MRT, where $\lambda_1 = 0.4$, $\lambda_2 = 0.5$ and there are 100 ants in the population.

IV. SIMULATION RESULTS AND DISCUSSIONS

In this section, we will demonstrate some simulation results to verify the proposed studies in this paper. If not specified, we consider that the size of the task is 250Mb and it is divided into 10 sub-tasks randomly. The user has a local CPU with a computational capability of 1×10^9 (cycle/s), and its local computational power is set to 2. There are three CAPs with a powerful computational capability with $M = 3$, and the associated CPU frequencies are set to 2×10^9, 6×10^9 and 6×10^9 (cycle/s), respectively. The pricing at the three CAPs is set to 0.1, 0.2 and 0.3 for 1M bits, respectively. The user is equipped with three transmit antennas with $K = 3$, and its transmit power of the upload and download is set to 4, and 3, respectively. The transmission bandwidth of the wireless links between the user and CAPs is set to 100 MHz. As to the ACO algorithm, we set the number of ants equal to 30 and each ant is executed 30 iterations. The information heuristic factor and the except heuristic factor are both set to 1, and the information volatility factor is set to 0.4.

Figs. 2 and 3 demonstrate the performance of the ACO algorithm versus the ant iteration, where $\lambda_1 = 0.4$ and $\lambda_2 = 0.5$ and there are 100 ants in the population. Specifically, Fig. 2 and Fig. 3 are associate with the MRT and SC schemes, respectively. For comparison, the performance of the brute force (BF) algorithm is also plotted in these two figures for comparison. As observed from these two figures, we can find that the system cost of the ACO algorithm decreases with the number of iteration, and after some iterations, the ACO algorithm becomes convergent. After convergence, the ACO algorithm achieves a near-optimal performance which is quite close to the BF algorithm. This validates the effectiveness of the proposed ACO algorithm for the offloading strategy in the MEC networks.

Figs. 4-5 show the effect of weight factors on the performance of ACO with MRT, where the number of ants in the ACO is 100 and the number of iterations is 100. Specifically, Fig. 4 corresponds to the effect of λ_1 which varies from

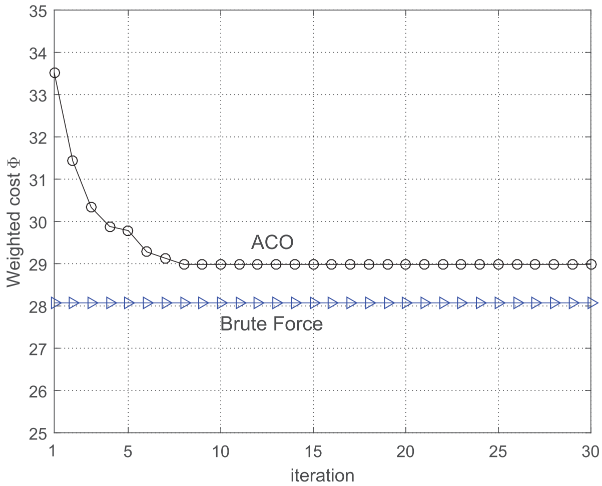


FIGURE 3. Performance of the ACO algorithm with SC, where $\lambda_1 = 0.4$, $\lambda_2 = 0.5$ and there are 100 ants in the population.

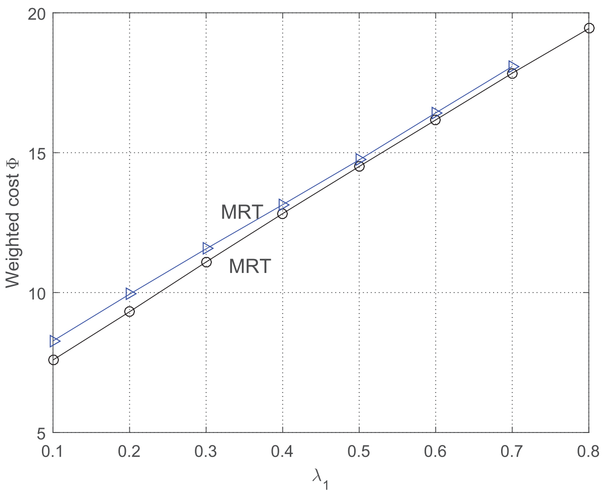


FIGURE 4. Effect of λ_1 on the performance of ACO with MRT, where $\lambda_2 = 0.1, 0.2$.

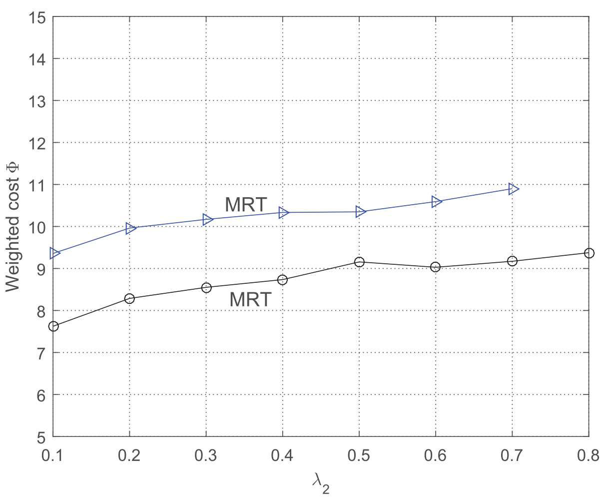


FIGURE 5. Effect of λ_2 on the performance of ACO with MRT, where $\lambda_1 = 0.1, 0.2$.

0.1 to 0.8, while Fig. 5 is associated with the effect of λ_2 which varies from 0.1 to 0.8. We can find from these two

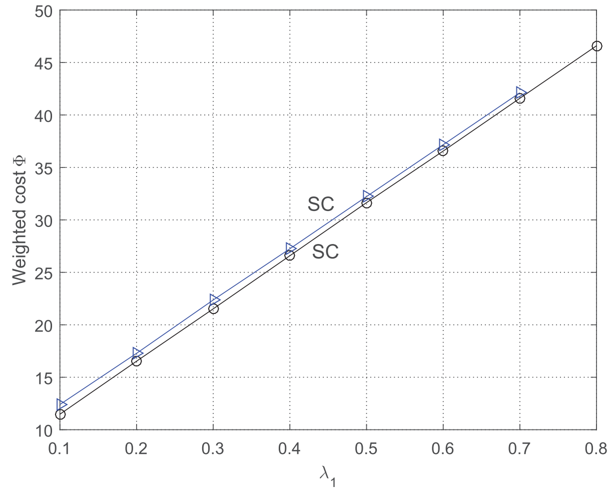


FIGURE 6. Effect of λ_1 on the performance of ACO with SC, where $\lambda_2 = 0.1, 0.2$.

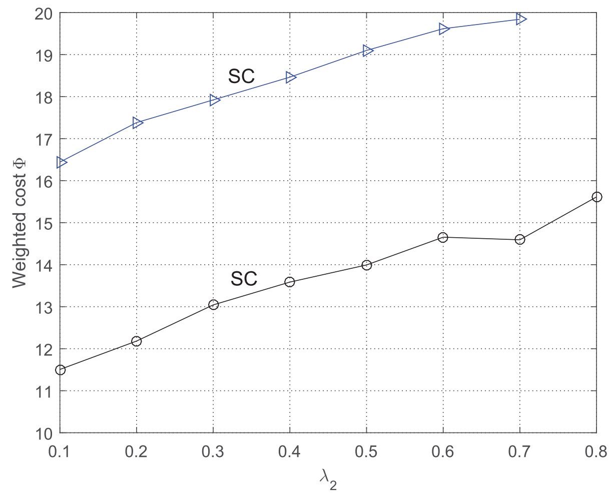


FIGURE 7. Effect of λ_2 on the performance of ACO with SC, where $\lambda_1 = 0.1, 0.2$.

figures that the weight factor λ_1 has an important impact on the system cost, indicating that the energy consumption is a dominant factor in the system whole cost. On the contrary, the system cost changes very limited with the change of λ_2, indicating that the latency is not dominant in the system whole performance.

Similar to Figs. 4-5, Figs. 6-7 depict the effect of weight factors λ_1 and λ_2 on the performance of ACO with SC, where the number of ants in the ACO is 100 and the number of iterations is 100. Specifically, Fig. 6 is associated with the effect of λ_1 which varies from 0.1 to 0.8, while Fig. 7 correspond to the effect of λ_2 which varies from 0.1 to 0.8. Similar to the observations in Figs. 4-5, we can find from Figs. 6-7 that the weight factor λ_1 has a profound impact on the system cost, indicating that the energy consumption is a dominant factor in the system whole cost. On the contrary, the system cost changes very limited with the change of λ_2, indicating that the latency is not dominant in the system whole performance.

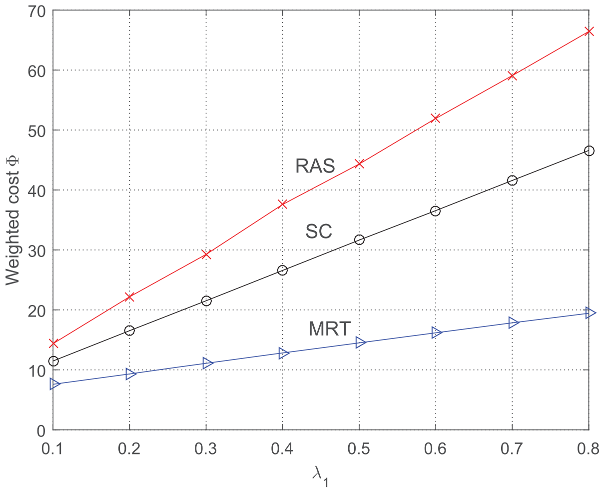


FIGURE 8. **Performance comparison of several array signal processing schemes versus λ_1, where $\lambda_2 = 0.1$ and there are 100 ants in the population.**

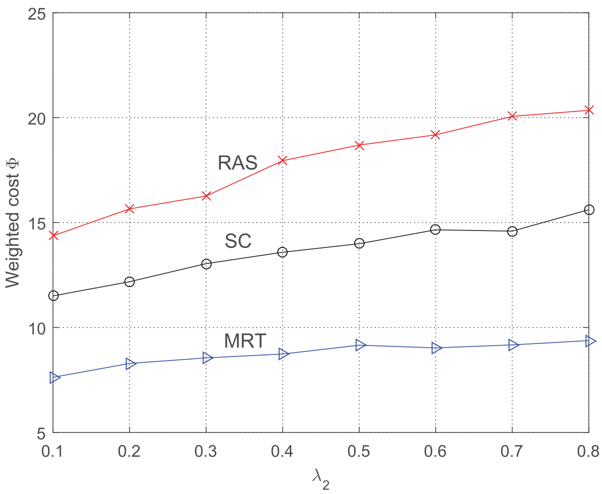


FIGURE 9. **Performance comparison of several array signal processing schemes versus λ_2, where $\lambda_1 = 0.1$ and there are 100 ants in the population.**

Figs. (8) and (9) compare the performances of several array signal processing schemes versus the weight factors, where the RAS, SC and MRT schemes are compared. In particular, Fig. 8 is associated with the performance versus the weight factor λ_1 with λ_2=0.1, while Fig. 9 correspond to the performance versus the weight factor λ_2 with λ_1=0.1. We can observe from these two figures that λ_1 has a more important role in the system performance than λ_2, indicating that the energy consumption dominates in the system whole cost. Moreover, the MRT has the lowest cost, while the SC outperforms the RAS in the system cost. This indicates that the MRT can exploit the gain from multiple antennas fully, while RAS fails to obtain the gain from multiple antennas. In further, the performance gap among the MRT, SC and RAS increases obviously with the larger value of λ_1, while it increases slightly with the large value of λ_2. This further validates the insight that the energy consumption plays a more important role in the system while cost.

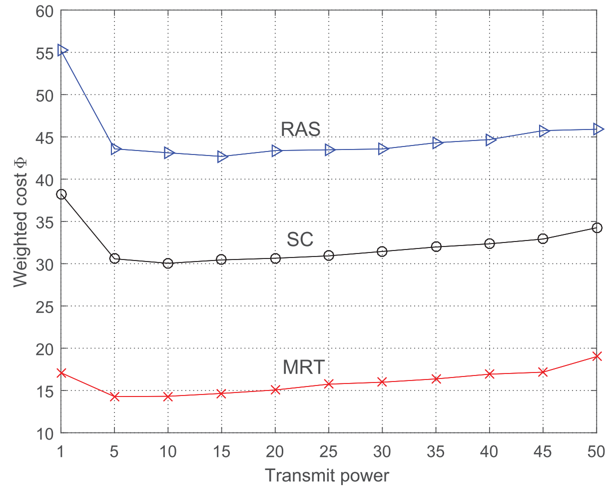


FIGURE 10. **Performance comparison of several array signal processing schemes versus the transmit power, where $\lambda_1 = \lambda_2 = 0.45$ and there are 100 ants in the population and 100 iterations in the ACO.**

Fig. 10 illustrates the performances of several array signal processing schemes versus the transmit power, where $\lambda_1 = \lambda_2 = 0.45$ and there are 100 ants in the population and 100 iterations in the ACO algorithm. In particular, the RAS, SC and MRT schemes are compared in Fig. 10. By observing Fig. 10, we can find that for each array signal processing scheme, the system cost firstly decreases with the increasing transmit power, and then instead increases with the transmit power when the transmit power is high. This is because that in the low region of transmit power, the increase of transmit power can help reduce the transmission latency effectively, while in the high region of transmit power, the increase of transmit power however imposes a severe load on the system energy consumption. Moreover, the result in Fig. 10 also shows that the MRT outperforms SC and RAS, while SC is better than the RAS in the terms of system cost. This further indicates that the array signal processing scheme has a significant impact on the system performance of the MEC networks.

V. CONCLUSION

In this paper, we studied the intelligent offloading strategy and bandwidth allocation for the MEC networks, where one user had some computational tasks to be computed either by the user itself or by the nearby CAPs in the networks at the cost of wireless transmission. The system cost was a linear combination of the computational price, energy consumption and latency. Based on the system cost, the intelligent offloading strategy was designed through the ACO algorithm. To further reduce the system cost, two array signal processing schemes were proposed to optimize the wireless transmission from the user to the CAPs. Simulation results were finally provided to show the effectiveness of the proposed ACO-based offloading strategy and the array signal processing schemes. In future works, we will consider the application for IoT networks, such as urban environment detection [44]–[46]. Moreover, we will investigate other kinds of intelligent algorithms [47]–[51] to the considered system, in order to further enhance the system performance.

REFERENCES

[1] H. Wang, L. Wan, M. Dong, K. Ota, and X. Wang, "Assistant vehicle localization based on three collaborative base stations via SBL-based robust DOA estimation," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5766–5777, Jun. 2019.

[2] X. Wang, L. Wan, M. Huang, C. Shen, and K. Zhang, "Polarization channel estimation for circular and non-circular signals in massive MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 5, pp. 1001–1016, Sep. 2019.

[3] L. Wan, X. Kong, and F. Xia, "Joint range-Doppler-angle estimation for intelligent tracking of moving aerial targets," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1625–1636, Jun. 2018.

[4] X. Lin, J. Xia, and Z. Wang, "Probabilistic caching placement in UAV-assisted heterogeneous wireless networks," *Phys. Commun.*, vol. 33, pp. 54–61, Apr. 2019.

[5] J. Xia, "Cache-aided mobile edge computing for B5G wireless communication networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, p. 15, Dec. 2020.

[6] X. Wang, "Joint resource allocation for cognitive OFDM-NOMA systems with energy harvesting in green IoT," *IEEE Access*, to be published.

[7] X. Lai, W. Zou, D. Xie, X. Li, and L. Fan, "DF relaying networks with randomly distributed interferers," *IEEE Access*, vol. 5, pp. 18909–18917, 2017.

[8] W. Huang, "Multi-antenna processing based cache-aided relaying networks for B5G communications," *Phys. Commun.*, to be published.

[9] C. Li, "Cache-enabled physical-layer secure game against smart UAV-assisted attacks in B5G noma networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2020, no. 1, pp. 1–10.

[10] D. Deng, J. Xia, L. Fan, and X. Li, "Link selection in buffer-aided cooperative networks for green IoT," *IEEE Access*, to be published.

[11] S. Pan, Z. Liu, and W. Lu, "Synthesis of naked plasmonic/magnetic Au/Fe_3O_4 nanostructures by plasmon-driven anti-replacement reaction," *Nanotechnology*, vol. 30, no. 6, Feb. 2019, Art. no. 065605.

[12] Y. K. Wang, Z. M. Xie, M. M. Wang, H. W. Deng, J. F. Yang, Y. Jiang, T. Zhang, X. P. Wang, Q. F. Fang, and C. S. Liu, "The superior thermal stability and tensile properties of hot rolled W-HfC alloys," *Int. J. Refractory Metals Hard Mater.*, vol. 81, pp. 42–48, Jun. 2019.

[13] S. Pan, X. Zhang, W. Lu, and S. F. Yu, "Plasmon-engineered anti-replacement synthesis of naked Cu nanoclusters with ultrahigh electrocatalytic activity," *J. Mater. Chem. A*, vol. 6, no. 38, pp. 18687–18693, Sep. 2018.

[14] M. M. Wang, Z. M. Xie, H. W. Deng, J. F. Yang, Y. K. Wang, T. Zhang, Y. Xiong, X. P. Wang, Q. F. Fang, and C. S. Liu, "Grain size effects of tungsten powder on the micro-structure and mechanical properties of tungsten-based alloys," *Mater. Sci. Eng., A*, vol. 754, pp. 216–223, Apr. 2019.

[15] X. Lai, L. Fan, X. Lei, J. Li, N. Yang, and G. K. Karagiannidis, "Distributed secure switch-and-stay combining over correlated fading channels," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 8, pp. 2088–2101, Aug. 2019.

[16] J. Yang, D. Ruan, J. Huang, X. Kang, and Y.-Q. Shi, "An embedding cost learning framework using GAN," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 839–851, 2020.

[17] F. Shi, J. Xia, Z. Na, X. Liu, Y. Ding, and Z. Wang, "Secure probabilistic caching in random multi-user multi-UAV relay networks," *Phys. Commun.*, vol. 32, pp. 31–40, Feb. 2019.

[18] X. Lin, Y. Tang, X. Lei, J. Xia, Q. Zhou, H. Wu, and L. Fan, "MARL-based distributed cache placement for wireless networks," *IEEE Access*, vol. 7, pp. 62606–62615, 2019.

[19] L. Fan, N. Zhao, X. Lei, Q. Chen, N. Yang, and G. K. Karagiannidis, "Outage probability and optimal cache placement for multiple amplify-and-forward relay networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12373–12378, Dec. 2018.

[20] J. Xia, L. Fan, W. Xu, X. Lei, X. Chen, G. K. Karagiannidis, and A. Nallanathan, "Secure cache-aided multi-relay networks in the presence of multiple eavesdroppers," *IEEE Trans. Commun.*, vol. 67, no. 11, pp. 7672–7685, Nov. 2019.

[21] B. Lu, "Interference suppression by exploiting wireless cache in relaying networks for B5G communications," *Phys. Commun.*, to be published.

[22] J. Xia, D. Deng, Y. Rao, D. Li, F. Zhu, and L. Fan, "When distributed switch-and-stay combining meets buffer in IoT relaying networks," *Phys. Commun.*, vol. 38, Feb. 2020, Art. no. 100920.

[23] Z. Zhao, R. Zhao, J. Xia, X. Lei, D. Li, C. Yuen, and L. Fan, "A novel framework of three-hierarchical offloading optimization for MEC in industrial IoT networks," *IEEE Trans. Ind. Informat.*, to be published.

[24] G. Liu, Y. Xu, Z. He, Y. Rao, J. Xia, and L. Fan, "Deep learning-based channel prediction for edge computing networks toward intelligent connected vehicles," *IEEE Access*, vol. 7, pp. 114487–114495, 2019.

[25] K. He, Z. Wang, W. Huang, D. Deng, J. Xia, and L. Fan, "Generic deep learning based linear detectors for MIMO systems over correlated noise environments," *IEEE Access*, to be published.

[26] S. Lai, "Intelligent secure communication for cognitive networks with multiple primary transmit power," *IEEE Access*, to be published.

[27] K. He, "A MIMO detector with deep learning in the presence of correlated interference," *IEEE Trans. Veh. Technol.*, to be published.

[28] C. Li, Y. Xu, J. Xia, and J. Zhao, "Protecting secure communication under UAV smart attack with imperfect channel estimation," *IEEE Access*, vol. 6, pp. 76395–76401, 2018.

[29] Y. Xu, J. Xia, H. Wu, and L. Fan, "Q-learning based physical-layer secure game against multiagent attacks," *IEEE Access*, vol. 7, pp. 49212–49222, 2019.

[30] C. Li, W. Zhou, K. Yu, L. Fan, and J. Xia, "Enhanced secure transmission against intelligent attacks," *IEEE Access*, vol. 7, pp. 53596–53602, 2019.

[31] J. Xia, Y. Xu, D. Deng, Q. Zhou, and L. Fan, "Intelligent secure communication for Internet of Things with statistical channel state information of attacker," *IEEE Access*, vol. 7, pp. 144481–144488, 2019.

[32] Z. Na, Y. Wang, X. Li, J. Xia, X. Liu, M. Xiong, and W. Lu, "Subcarrier allocation based simultaneous wireless information and power transfer algorithm in 5G cooperative OFDM communication systems," *Phys. Commun.*, vol. 29, pp. 164–170, Aug. 2018.

[33] J. Zhao, X. Guan, and X. P. Li, "Power allocation based on genetic simulated annealing algorithm in cognitive radio networks," *Chin. J. Electron.*, vol. 22, no. 1, pp. 177–180, Jan. 2013.

[34] B. Wang, F. Gao, S. Jin, H. Lin, and G. Y. Li, "Spatial- and frequency-wideband effects in millimeter-wave massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 66, no. 13, pp. 3393–3406, Jul. 2018.

[35] C. Lu, W. Xu, S. Jin, and K. Wang, "Bit-level optimized neural network for multi-antenna channel quantization," *IEEE Wireless Commun. Lett.*, vol. 9, no. 1, pp. 87–90, Jan. 2020.

[36] Z. Na, J. Lv, M. Zhang, B. Peng, M. Xiong, and M. Guan, "GFDM based wireless powered communication for cooperative relay system," *IEEE Access*, vol. 7, pp. 50971–50979, 2019.

[37] J. Zhao, S. Ni, L. Yang, Z. Zhang, Y. Gong, and X. You, "Multiband cooperation for 5G HetNets: A promising network paradigm," *IEEE Veh. Technol. Mag.*, vol. 14, no. 4, pp. 85–93, Dec. 2019.

[38] H. Xie, F. Gao, S. Zhang, and S. Jin, "A unified transmission strategy for TDD/FDD massive MIMO systems with spatial basis expansion model," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3170–3184, Apr. 2017.

[39] X. Hu, C. Zhong, X. Chen, W. Xu, and Z. Zhang, "Cluster grouping and power control for angle-domain MmWave MIMO NOMA systems," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 5, pp. 1167–1180, Sep. 2019.

[40] J. Zhao, Q. Li, Y. Gong, and K. Zhang, "Computation offloading and resource allocation for cloud assisted mobile edge computing in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7944–7956, Aug. 2019.

[41] W. Xu, J. Liu, S. Jin, and X. Dong, "Spectral and energy efficiency of multi-pair massive MIMO relay network with hybrid processing," *IEEE Trans. Commun.*, vol. 65, no. 9, pp. 3794–3809, Sep. 2017.

[42] Z. Junhui, Y. Tao, G. Yi, W. Jiao, and F. Lei, "Power control algorithm of cognitive radio based on non-cooperative game theory," *China Commun.*, vol. 10, no. 11, pp. 143–154, Nov. 2013.

[43] S. Ni, J. Zhao, H. H. Yang, and Y. Gong, "Enhancing downlink transmission in MIMO HetNet with wireless backhaul," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 6817–6832, Jul. 2019.

[44] J. Yang, Z. Lin, H. Wu, Q. Chen, X. Xu, G. Huang, L. Fan, X. Shen, and K. Gan, "Inverse optimization of building thermal resistance and capacitance for minimizing air conditioning loads," *Renew. Energy*, vol. 148, pp. 975–986, Apr. 2020.

[45] J. Yang, H. Wu, X. Xu, G. Huang, T. Xu, S. Guo, and Y. Liang, "Numerical and experimental study on the thermal performance of aerogel insulating panels for building energy efficiency," *Renew. Energy*, vol. 138, pp. 445–457, Aug. 2019.

[46] H. Huang, Y. Zhou, R. Huang, H. Wu, Y. Sun, G. Huang, and T. Xu, ''Optimum insulation thicknesses and energy conservation of building thermal insulation materials in Chinese zone of humid subtropical climate,'' *Sustain. Cities Soc.*, vol. 52, Jan. 2020, Art. no. 101840.
[47] R. Zhao, ''Deep reinforcement learning based mobile edge computing for intelligent Internet of Things,'' *IEEE Access*, to be published.
[48] C. Fan and Y. Ding, ''Cooling load prediction and optimal operation of HVAC systems using a multiple nonlinear regression model,'' *Energy Buildings*, vol. 197, pp. 7–17, Aug. 2019.
[49] Z. Zhao, ''Intelligent mobile edge computing with pricing in Internet of Things,'' *IEEE Access*, to be published.
[50] C. Fan, Y. Ding, and Y. Liao, ''Analysis of hourly cooling load prediction accuracy with data-mining approaches on different training time scales,'' *Sustain. Cities Soc.*, vol. 51, Nov. 2019, Art. no. 101717.
[51] J. Xia and D. Deng, ''A note on implementation methodologies of deep learning-based signal detection for conventional MIMO transmitters,'' *IEEE Trans. Broadcast.*, to be published.

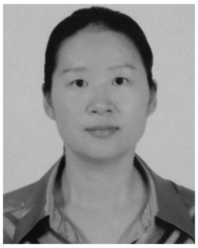
YINGHAO GUO received the bachelor's degree in software engineering from Zhengzhou University of Education, in June 2019. He is currently pursuing the degree with the School of Computer Science and Cyber Engineering, Guangzhou University. His current research interests include machine learning and mobile edge computing resource scheduling algorithms.

ZICHAO ZHAO received the B.E. degree in computer science and technology from Yantai University, Yantai, China, in 2018. He is currently pursuing the M.E. degree with Guangzhou University, Guangzhou, China. His current research interests include mobile edge computing.

RUI ZHAO received the B.E. degree in computer science and technology from Bohai University, Jinzhou, China, in 2018. He is currently pursuing the M.S. degree with Guangzhou University, Guangzhou, China. His current research interests include machine learning and mobile edge computing resource scheduling algorithms.

SHIWEI LAI received the bachelor's degree in computer science and technology from the Guangdong University of Education, in June 2019. She is currently pursuing the degree with the School of Computer Science and Cyber Engineering, Guangzhou University. Her current research interests include machine learning and mobile edge computing resource scheduling algorithms.

ZOU DAN received the M.S. degree from East China Jiaotong University, in 2008, where she is currently pursuing the Ph.D. degree. Her research interests include vehicular networks, mobile edge computing, and resource allocation.



JUNJUAN XIA received the bachelor's degree from the Department of Computer Science, Tianjin University, in 2003, and the master's degree from the Department of Electronic Engineering, Shantou University, in 2015. She is currently working as a Laboratory Assistant with the School of Computer Science and Cyber Engineering, Guangzhou University. Her current research interests include wireless caching, physical-layer security, cooperative relaying, and interference modeling.

LISENG FAN received the Ph.D. degree from the Tokyo Institute of Technology, Tokyo, in 2008. He is currently a Professor with the School of Computer Science and Cyber Engineering, Guangzhou University. He has published more than 40 articles on the IEEE journal and IEEE conferences. His main research interests include the information security, wireless networks, and the artificial intelligence. His recent research interest is the application of artificial intelligence into the wireless networks.