IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 14, 2021, accepted June 23, 2021, date of publication July 1, 2021, date of current version July 13, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3093886

Intelligent Mobile Edge Computing Networks for Internet of Things

LIMING CHEN[1], XIAOYUN KUANG[1], (Member, IEEE), FUSHENG ZHU[2], AND JUNJUAN XIA[3]

[1]Electric Power Research Institute of China Southern Power Grid, Guangzhou 511483, China
[2]Guangdong New Generation Communication and Network Innovative Institute (GDCNi), Guangzhou 510700, China
[3]School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou 510006, China

Corresponding authors: Liming Chen (chenlm1018@foxmail.com), Fusheng Zhu (zhufusheng@gdcni.cn), and Junjuan Xia (xiajunjuan@gzhu.edu.cn)

This work was supported in part by the Key-Area Research and Development Program of Guangdong Province under Grant 2018B010124001, in part by the Science & Technology Projects of China Southern Power Gird under Grant SEPRI-K205020, in part by the International Science and Technology Cooperation Projects of Guangdong Province under Grant 2020A0505100060, in part by the Natural Science Foundation of Guangdong Province under Grant 2021A1515011392, and in part by the Research Program of Guangzhou University under Grant YK2020008/YJ2021003.

ABSTRACT In this work, an intelligent mobile edge computing (MEC) network is studied for Internet of Things (IoT) in the presence of eavesdropping environments, where there are multiple users who can offload their confidential tasks to the computational access point (CAP) for the assistance of computation. One unmanned aerial vehicle (UAV) attacker exists in the system and it can listen to the confidential data transmission from the users to the CAP. We optimize the system design of the intelligent MEC network, by adaptively allocating the offloading ratio and wireless bandwidth, to reduce the linearly weighted cost of the latency as well as energy consumption (EnC). Specifically, starting from the deep reinforcement learning, we devise a deep Q-network (DQN) network to adjust the offloading ratio and transmission bandwidth, which can help calculate the computational tasks and suppress the eavesdropping from the UAV efficiently. We finally provide some simulation results to validate the proposed offloading strategy. In particular, the proposed offloading strategy can achieve a much lower cost compared to the conventional ones, in the terms of latency and EnC.

INDEX TERMS Deep reinforcement learning, Internet of Things, mobile edge computing, task offloading, unmanned aerial vehicles.

I. INTRODUCTION

In recent years, with the rapid development of transmission technology, the data rate of wireless networks has increased significantly [1]–[3], which has promoted the development of Internet of Things (IoT) [4], [5]. What follows is a huge amount of access points and traffic data, as well as highly intensive transmission and computational tasks, which undoubtedly put forward a much higher requirement on the wireless networks [6]–[8]. Therefore, the current research on the wireless networks has gradually changed from only the mode of communication and transmission to both wireless transmission and task calculation [9]–[11]. For example, in application scenarios such as smart monitoring systems, smart transportation systems, and Internet of Vehicles, mobile devices such as smart phones or smart cars need to continuously receive data and process computational tasks. However, the computational capability of mobile devices is very limited, and processing large-scale data will cause a lot of system latency, which will seriously reduce the quality of user services [12]–[14]. To deal with this problem, researchers have proposed cloud computing solutions, where the computational tasks of mobile device can be offloaded to the cloud server for the assistance of computation, and then the results will be fedback to the mobile devices. The cloud computing solution is to use the powerful computational capability of the cloud server to make up for the lack of computational capability of the local devices, which reduces the system latency

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

to a certain extent. However, the energy consumption (EnC) and latency caused by the transmission of the computational tasks offload to the cloud server cannot be ignored, and the computational tasks are sent to the remote cloud servers, which will cause the user's confidential information to be eavesdropped. Therefore, researchers proposed mobile edge computing (MEC) technique, which is an extension of cloud computing technology [15]–[17]. An accessible computational access point (CAP) is established at the edge of a mobile network, providing mobile devices with intelligent services, such as the computational resources and storage resources. Since the CAP is often close to mobile users, it can help reduce system EnC and latency, improve user experience, and meet the needs of mobile networks for agile connections. From the perspective of technological evolution, MEC technology is actually the sinking and extension of cloud computing technology [18], [19]. Considering the situation that massive amounts of data and traffic generated form Internet, low-latency response, and green communications and wider device connections, cloud computing technology has first begun to evolve into a decentralized distributed form, and then evolve into mobile edge computing technology. How to further improve the user service quality in the MEC networks and how to schedule and allocate various resources of distributed networks has become a hot topic that researchers are paying more attention to.

Reasonable resource scheduling and allocation are the keys to improve the network performance [20]–[22]. The allocation of communication spectrum resources and the control of task offloading ratio are the key design in the MEC networks [23], [24]. In this area, a spectrum allocation strategy and offloading strategy could be used for the MEC network based on the optimized analysis and game theory. Based on this strategy, mobile devices can perform communication spectrum resource scheduling and offloading ratio regulation, reducing the system EnC and latency, and improving user's service quality [25], [26]. However, in practice, the spectrum allocation problem and offloading strategy problem may be an NP-hard problem, due to dynamic change for a large number of mobile devices and the network environment, so that the conventional mathematical methods may be difficult to solve the optimization problem. With the rise of artificial intelligence technology, more and more researchers are currently focusing on using machine learning to solve complex optimization problems [27], [28]. Among them, the model-free reinforcement learning algorithm can be used to achieve the optimal strategy in solving many discrete and non-convex decision-making problems. This excellent performance makes reinforcement learning widely used for this type of decision-making problem. However, when the state space becomes huge and the agent can perform more actions, the complicated process of updating parameters makes the convergence of reinforcement learning slower. As a result, the reinforcement learning with deep neural networks can be used to design a deep reinforcement learning algorithm that can be applied to a large number of state space and action space scenarios in the MEC networks, which can achieve a faster convergence and better performance compared to the conventional reinforcement learning.

Although MEC technology provides the computational services, which enhances the transmission reliability to a certain extent, its transmission security is threatened easily by eavesdroppers due to the broadcast nature of wireless transmission. The physical-layer security of the wireless transmission can be studied to put forward a secure strategy implemented effectively for the wireless networks [29]–[31]. The antenna selection algorithm can be used to select the number of transmit antennas to combat the eavesdroppers in the wireless network and protect the physical-layer security. Aiming to reduce the transmit rate of the communication system, an intelligent attacker with multiple working modes is studied, where the attacker can perform four modes: eavesdropping, interference, deceiving, and keeping silent. If the network still uses the conventional physical-layer secure strategies, attacks from the intelligent attackers are hard to be suppressed. To solve this problem, a secure strategy based on reinforcement learning can be used. By intelligently regulating the transmit power of mobile users, the intelligent attack can be effectively suppressed, to improve the transmit rate of the wireless link, and increase the security of the physical-layer transmission. In the practical scenarios, the attacker's position may change dynamically, which makes the channel parameters of the eavesdropping link change, causing the existing power control algorithm no longer applicable [32]–[34]. So far, there has been little work on the protect of secure MEC networks, considering the existence of an intelligent eavesdropper with a changeable location. Therefore, the task offloading strategy and the spectrum allocation strategy should be jointly to minimize the system latency and EnC while fighting against the intelligent eavesdroppers, which motivates the work in this paper.

In this work, an intelligent MEC network is studied for IoT in the presence of eavesdropping environments, where there are multiple users who can offload their confidential tasks to the CAP for the assistance of computation. One UAV attacker exists in the system and it can listen to the confidential data transmission from the users to the CAP. We optimize the system design of the intelligent MEC network, by adaptively allocating the offloading ratio and wireless bandwidth, to reduce the linearly weighted cost of the latency as well as EnC. Specifically, starting from the deep reinforcement learning, we devise a deep Q-network (DQN) network to adjust the offloading ratio and transmission bandwidth, which can help calculate the computational tasks and suppress the eavesdropping from the UAV efficiently. We finally provide some simulation results to validate the proposed offloading strategy. In particular, the proposed offloading strategy can achieve a much lower cost compared to the conventional ones, in the terms of latency and EnC.

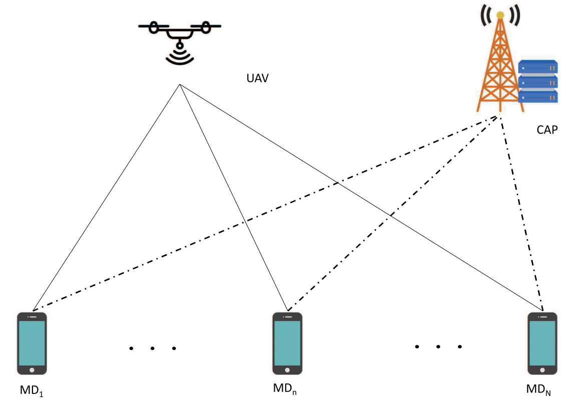

FIGURE 1. A multi-user MEC network in the eavesdropping environments.

II. SYSTEM MODEL

Fig. 1 depicts the system model of MEC networks in the presence of one eavesdropper, in which there is one CAP with a powerful computational capability, a UAV eavesdropper that moves all the time, and N users in the system. The user set is $\mathcal{N} = \{\mathrm{MD}_n | n = 1, 2, \ldots, N\}$. In each time slot, user S_n needs to compute a task of l_n bits. In this paper, the user can be a mobile device, such as a mobile phone or a smart wearable device. Therefore, the user S_n has some limitations in terms of computational capability and EnC. To reduce the latency and EnC come from the computational tasks, users should choose to offload some or all of the tasks to the nearby CAPs. At the same time, there is a drone attacker who moves at all times to overhear the confidential communication from the user to the CAP. Assuming that the task of user S_n can be separated into two portions arbitrarily, where one portion is executed locally, while the other portion is executed at the CAP and the computational result is fedback to the user. We use $\alpha_n \in [0, 1]$ to indicate the offload ratio of the n-th user. For user S_n, this means that there are $(1-\alpha_n) \cdot l_n$ tasks to be computed at local, while the residual $\alpha_n \cdot l_n$ tasks are offloaded to the CAP for computation.

A. SECURE TRANSMISSION MODEL

In this paper, the UAV eavesdroppers are considered in the system to overhear the confidential transmission of the link from the user to the CAP, where the problem is transformed into the study of the impact of the system secrecy data rate on the entire system. When the user S_n offloads a portion of the task, that is, $0 < \alpha_n \leq 1$, the sender needs to cope with the eavesdroppers while offloading the tasks due to the existence of eavesdroppers in the networks. When the channel between the user and the CAP is more advantageous than the channel between the user and the attacker, the confidential information can be transmitted by the legitimate users through a reasonable design of channel coding, modulation and other technologies. To begin with, we need to obtain the data rate of the main link, given by,

$$R_{n,C} = W_n \log_2\left(1 + \frac{P^n_{tran}|h_{n,C}|^2}{\sigma^2_{n,C}}\right), \tag{1}$$

where W_n is the wireless bandwidth of user S_n, P^n_{tran} is the transmit power of user S_n, $h_{n,C} \sim (0, \epsilon_{n,C})$ is the channel parameter of the link from user S_n to the CAP, and $\sigma^2_{n,C}$ is the variance of the additive white Gaussian noise (AWGN) at the CAP. If the eavesdropper chooses to overhear the user's confidential information, we can use Shannon's theory to obtain the data rate of the eavesdropping link as,

$$R_{n,E} = W_n \log_2\left(1 + \frac{P^n_{tran}|h_{n,E}|^2}{\sigma^2_{n,E}}\right) \tag{2}$$

where $h_{n,E} \sim (0, \epsilon_{n,E})$ is the channel parameter of the link from the user S_n to the eavesdropper, and $\sigma^2_{n,E}$ is the variance of AWGN at the UAV. From (1) and (2), we can write the secrecy data rate as,

$$R_n = \left[R_{n,C} - R_{n,E}\right]^+ \tag{3}$$

where $[X]^+ = X$ if $X > 0$, or zero otherwise. This shows that when the conditions of the secure transmission between the user and the CAP are satisfied, the user S_n can offload its computational tasks. On the other hand, when the conditions of secure transmission between the user S_n and CAP cannot be satisfied, that is, $R_{n,C} - R_{n,E} \leq 0$, the user S_n cannot offload its computational tasks to the CAP, that is, the offloading ratio $\alpha_n = 0$. Moreover, we can find that the offloading ratio affects the eavesdropping capability, as the attacker can overhear the confidential message during the offloading process. In particular, when the offloading ratio is zero indicating that the offloading process does not occur, the system computation will be guaranteed, at the cost of a large computational latency and energy consumption at local. In further, the channel parameters affect the eavesdropping capability as well, since the secrecy data rate is affected by the instantaneous channel parameters. In particular, when the eavesdropping channel gains become larger, the eavesdropping capability of the UAV attacker will increase.

B. LOCAL COMPUTATIONAL MODEL

The local computational model involves the latency and EnC required for tasks that are calculated locally by the user. Since the local calculation does not require wireless transmission, the UAV eavesdropper in the system will not affect the local computation. We use f_n to represent the computational capability of the user S_n (the number of cycles of the user's CPU per second). At this time, through the computational capability of the user S_n, the time to complete the local computational task is

$$T^n_{comp} = \frac{l_n}{f_n}(1 - \alpha_n)\omega, \tag{4}$$

where ω denotes CPU speed measured by the CPU cycles in computing one bit. Accordingly, the EnC to complete the local task is

$$E^n_{comp} = T^n_{comp} P^n_{comp}, \tag{5}$$

where P^n_{comp} denotes the computational power of S_n.

C. TASK OFFLOADING MODEL

As mentioned in the previous part, when the conditions for the secure transmission can be satisfied between the user S_n and the CAP, the user can offload some portions or all tasks for calculation. In this case, the transmission latency in offloading the tasks is,

$$T^n_{tran} = \frac{l_n \alpha_n}{R_n}. \tag{6}$$

Meanwhile, the transmission EnC is,

$$E^n_{tran} = T^n_{tran} P^n_{tran}. \tag{7}$$

After the user offloads its task to the CAP, the CAP calculates the task and feedbacks to the user. First, as to the CAP, the latency in calculating the offloaded tasks is,

$$T^C_{comp} = \frac{l_n \alpha_n}{F_C} \omega, \tag{8}$$

where F_C denotes the CAP's computational capability. In practical environments, the CAP is often connected to the power supply to provide stable and high-quality services. Therefore, we ignore the EnC of the processing at the CAP.

As to the most offloading tasks, such as face recognition, health assessment, and other services, the result file is very small, so that the time to feedback the result can be ignored. At this point, the latency and EnC have all been described for the secure MEC networks. Next, we need to summarize the latency and EnC coming from the transmission and computation. First, we can sum the latency of a computational task, the latency of the task offloading, and the latency of computing the offloaded tasks in the CAP, to obtain the total latency as,

$$T_{total} = \sum_{n=1}^{N} \left(T^n_{comp} + T^n_{tran} + T^C_{comp} \right). \tag{9}$$

At the same time, we can sum the EnC of calculating the local task and the EnC of the task offloading, to obtain the total EnC as,

$$E_{total} = \sum_{n=1}^{N} \left(E^n_{comp} + E^n_{tran} \right). \tag{10}$$

Note that The method of offloading some computational tasks is a very common solution in the MEC networks. By offloading some tasks to the edge computational nodes, the latency and EnC of mobile devices to complete the tasks can be reduced. It is also worth notable that the optimization problem is a multi-objective one, as both the latency and EnC are involved. When the latency and EnC are reduced to a certain degree, they will form a state of mutual restraint, that is, reducing EnC will be accompanied by an increase in the latency meanwhile. To deal with different scenarios, we design a weight factor λ to weigh the latency and EnC as,

$$\Phi = \lambda T_{total} + (1 - \lambda) E_{total} \tag{11}$$

where the notation Φ denotes the overall optimization goal of the system, and the weight factor λ is in the range of [0, 1].

TABLE 1. Symbol description.

Symbol	Definition
MD_n	Number of local users
CAP	Computational access point
l_n	Computational tasks of user S_n
α_n	Offloading ratio of user S_n
$R_{n,C}$	Transmit rate between S_n and CAP
$R_{n,E}$	Transmit rate between S_n and eavesdropper
R_n	Secrecy data rate
T^n_{comp}, E^n_{comp}	Computational latency and EnC locally
T^n_{tran}, E^n_{tran}	Transmit latency and EnC
T^C_{comp}	Computational latency of the processing at the CAP
T^C_{comp}, E_{total}	System latency and EnC

The introduction of weight factor allows us to adjust the importance of the latency and EnC. If the system has a high requirement on the low latency, we need to enlarge λ. On the contrary, the value of λ should be decreased to increase the importance of EnC in the entire system. The introduction of weight factor can also enable us to have a more comprehensive understanding of the trade-off mechanism between the latency and EnC in the MEC networks.

From (11), we obtain the optimization objective function Φ in the system, and minimizing Φ is the key to optimize the entire system. Many factors affect the objective function, such as the bandwidth optimization, the design of transmit power, and the way to counter the eavesdropping channels to enhance the secrecy data rate during the task offloading. Then, the impact factor is transformed into reducing Φ, or by designing the offloading strategy $\alpha_n \in [0, 1]$ to improve the utilization of computational resources to improve system performance. This paper mainly studies the impact of the task offloading strategy and the bandwidth when the user communicates with the CAP. Accordingly, we can design the network by minimizing the system joint cost Φ, through designing the offloading strategy and bandwidth allocation,

$$\begin{aligned} \min_{\{\alpha_n, W_n\}} \quad & \Phi, \\ \text{s.t. } \mathrm{C}_1 : & \ \alpha_n \in [0, 1], \\ \mathrm{C}_2 : & \sum_{n \in \mathcal{N}} W_n = W_{total}, \end{aligned} \tag{12}$$

where W_{total} is the total bandwidth of the system. The symbols used in this paper are summarized in Table 1.

III. TASK OFFLOADING ALGORITHM BASED ON REINFORCEMENT DEEP LEARNING

Sec. II introduced the task offloading and bandwidth allocation models in the secure MEC networks, and proposed that the main goal of this work is to reduce the system joint cost Φ as much as possible. To this end, we will firstly solve the problem of the task offloading among users, and then dynamically optimize the bandwidth allocation according to the offloading ratio under the condition of eavesdropping channels. We further use the deep reinforcement learning, in order to address the problem of the task offloading and bandwidth allocation.

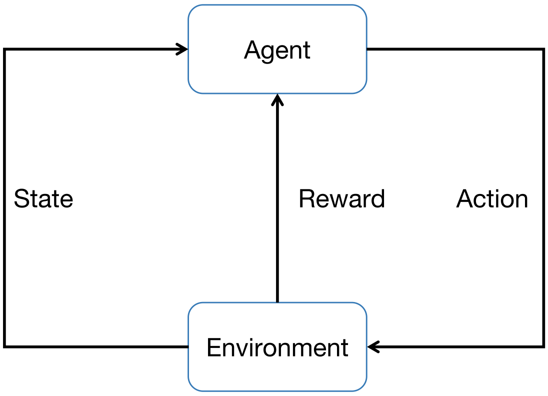

FIGURE 2. Reinforcement learning process.

It is difficult for the conventional optimization to deal with the resource allocation and bandwidth allocation under the secure MEC networks, especially when there is an eavesdropper dynamically overhearing the wireless transmission. Fortunately, the reinforcement learning technology can perform well to solve the dynamic optimization problems, which is an ideal technology for optimizing the task offloading in the secure MEC networks.

A. REINFORCEMENT LEARNING

Fig. 2 illustrates the reinforcement learning process, which is composed of one agent as well as the associated environment. In this scenario, the CAP can be the agent, while the other nodes can regarded as the environment. The agent can make some decisions through looking at the changes in state. As to the reinforcement learning, how to design an appropriate reward function plays an important role. For the current state, it is very important to reward the appropriate behavior, which is designed as,

$$r=\begin{cases}1, & \text{If } (\Phi_t-\Phi_{t-1})>0\\ -1, & \text{Else,}\end{cases} \tag{13}$$

where Φ_t represents the system cost at time t. We can model the task offloading process through a Markov decision process (MDP), where S is the state space, each s in S belongs to the description of the perceived environment, and A is the action space. If a specific action acts on the state s, it will occur. State transition s_t is transited to s_{t+1}, and the reward r is returned.

B. DEEP REINFORCEMENT LEARNING

The conventional reinforcement learning based schemes, such as Q-learning, often use tables to preserve state value functions. However, for the task offloading and bandwidth allocation problems, we cannot preserve the value function in a table, when the state dimension is too large. In this paper, as the state space is too large, we use the deep reinforcement learning to address the problem of task offloading for the considered system. Compared to the conventional reinforcement learning, DQN employs a parameterized neural network, which serves as the approximator of a value function to deal with the problem of task offloading and bandwidth allocation. Fig. 3 depicts the structure of the deep neural network, where we use S to denote the input of the neural network and output the next action a to be taken. The neural network consists of an input layer and three hidden layers. To obtain the next action, a greedy selection strategy is set at the output of the neural network, and use a probability ϵ to choose one random action a, described by,

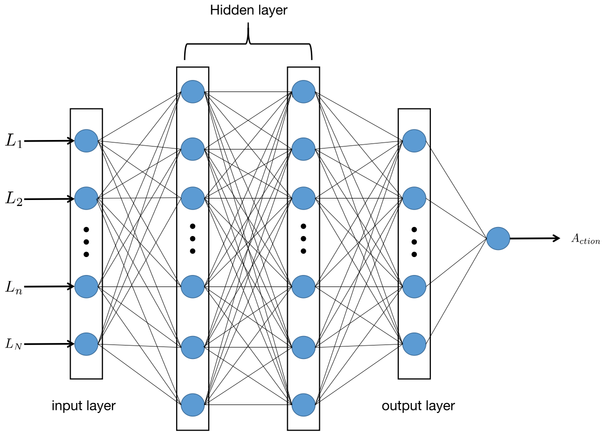

FIGURE 3. Structure of the deep neural network.

$$A=\begin{cases}random, & \epsilon\\ \arg\min_a(Q(s_t,a;\omega)), & 1-\epsilon\end{cases} \tag{14}$$

The interaction sequence in the reinforcement learning has a certain correlation among states. If we train the neural network directly, the training performance may be poor due to the correlation. To solve the correlation issue,

we adopt the experience replay unit, where the experience replay unit includes both the collecting and sampling samples. The network randomly selects a batch of samples among the cache to train, so that we can obtain a more stable training result. Meanwhile, we train a sample by multiple times, which helps improve the sample utilization rate.

To implement the D-DQN based offloading strategy, we employ the target network (TargetNet), where the same parameters exist in the two models before the start of training. The loss function of DQN is defined as the difference between the target value Q_{target} and the predicted value $Q(s, a:\omega)$,

$$Loss_\omega=(Q_t-Q(s,a:\omega))^2, \tag{15}$$

with

$$Q_t=(r-\gamma\arg\min_a(Q(s',a';\omega'))). \tag{16}$$

Therefore, we can fix the target value within a period in the target network, and finally the volatility of the model is reduced.

We summarize the whole process of the DQN-based offloading strategy in Algorithm 1, where the flow chart is given as,

Algorithm 1 DQN-Based Resource Allocation Optimization for the MEC Network

```
Input: User task $i$, total bandwidth $W_{total}$ user and CAP computational capability.
Output: Task offloading and bandwidth allocation results.
Initialize replay memory $D$ to capacity $N$,
Use $\omega$ to initialize the weight of the action value function $Q$, Use $\omega'/=\omega$ to initialize the target action value function.
Initialization state $S_1$,
    for t = 1: $T$ do
        Randomly select action $a_t$ by probability $\epsilon$.
        $a_t = \begin{cases} random, & \epsilon \\ \arg\min_a(Q(s_t, a; w)), & 1-\epsilon \end{cases}$
        Perform the action $a_t$ in the environment, observe the reward $r$ and the next state $s_{t+1}$.
        Store relevant parameters $(s_t, a_t, r_t, s_{t+1})$ in $D$.
        Randomly select a small batch $(s_i, a_i, r_i, s_{i+1})$ for training in $D$.
        $y_i = r_i + \text{argmin}_a(\hat{Q}(s'_t, a'; \omega'))$.
        Perform gradient descent.
        Update every $C$ step $\hat{Q} = Q$.
    end for
```

- Approximate the value of Q by using a neural network parameterized by ω,
- Define the loss function by using the MSE of the Q value,
- Compute the gradient of the loss function parameterized by ω,
- Optimize the parameters by using the stochastic gradient descent (SGD).

C. TASK OFFLOADING STRATEGY BASED ON DQN

This paper uses a proportion of offloading tasks. This is because when multiple users exist, it is difficult for the conventional methods to address such kind of problem. We hence use the deep reinforcement learning to devise an offloading strategy, where the state space is defined as $S = \{l_1\alpha_{1,m}, l_2\alpha_{2,m}, \ldots, l_N\alpha_{N,m}\}$, and the action space is defined as $A = \{\alpha_{1,m}, \alpha_{2,m}, \ldots, \alpha_{N,m}\}$.

D. BANDWIDTH ALLOCATION STRATEGY BASED ON DQN

After the task offloading ratio is determined, this paper investigates that the eavesdropper attacks the link between the user and the CAP during the task offloading process, which will affect the system secrecy data rate as well as the allocated bandwidth. To safeguard the task offloading, we design a bandwidth allocation strategy based on DQN. In (3) and (4), $R_{n,C}$ and $R_{n,E}$ are continuously changing. By dynamically allocating bandwidth W_n for the user S_n, the system cost Φ can be further reduced. A DQN-based bandwidth allocation strategy is proposed in Fig. 4, where the main steps are as follows,

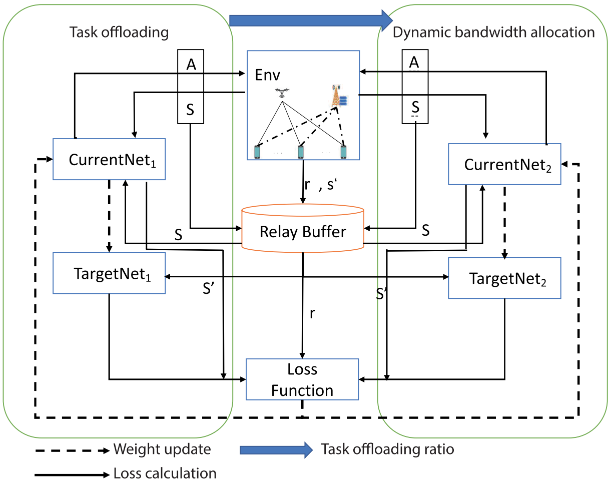

FIGURE 4. Proposed offloading and bandwidth optimization scheme.

- Obtain the offloading ratio of each mobile device through the DQN-based task offloading strategy in an environment with a fixed bandwidth,
- After obtaining the task offloading ratio, the system starts to offload the task, and the wireless channel is attacked by eavesdroppers during the offloading process,
- We adopt the bandwidth allocation strategy based on DQN to allocate the bandwidth among users.

IV. SIMULATION RESULTS AND DISCUSSIONS

In this section, we evaluate the proposed D-DQN based offloading strategy with the conventional ones, in terms of latency and EnC. The wireless channels in the system are subject to the Rayleigh flat fading [35]–[38], where the average channels gains of the main and eavesdropping links are set to 0.9 and 0.2, respectively. The transmit power at the users is 2W, while the associated computational power is 3W. For the implementation of the proposed DQN method, we use the TensorFlow 1.14.0 platform with Python 3.6.8, to perform the training and test processes. In the deep neural network, there are two hidden layers. The CAP allocates the same computational capability to the users. In addition, there are 5 users in the system, whose computational capabilities are $[1.4, 0.21, 0.95, 0.13, 0.43] \times 10^8$ cycle/sec.

Fig. 5 shows the convergence of the D-DQN based scheme with respect to the iteration, where the iteration varies from 0 to 12000. For comparison, the weighted cost of local computation only (LCO) and computation at the CAP only (CCO) are also plotted in Fig. 5, where the computational tasks are totally computed at local and by the CAP, respectively. We observe from Fig. 5 that after some enough iterations, the proposed D-DQN based scheme becomes convergent. In particular, the proposed scheme is convergent after about 8000 iterations. Moreover, the proposed scheme outperform the conventional LCO and CCO, as it can jointly exploit the computational resources coming from the users and CAP.

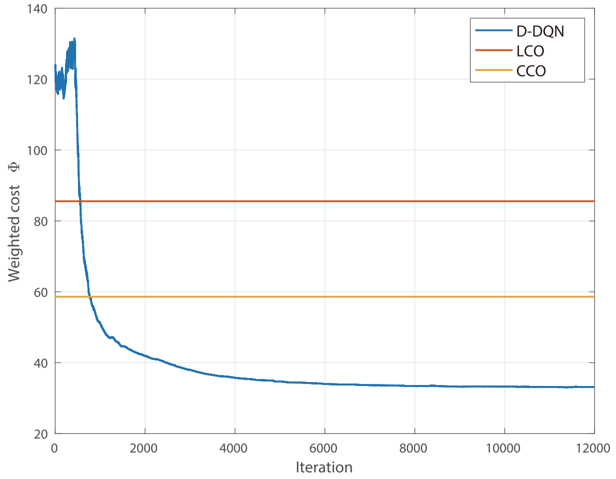

FIGURE 5. Convergence of the D-DQN based scheme.

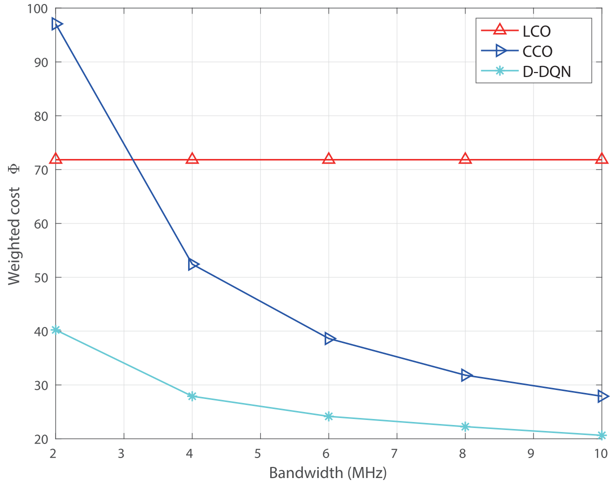

FIGURE 6. Comparison of three offloading strategies with respect to W_{total}.

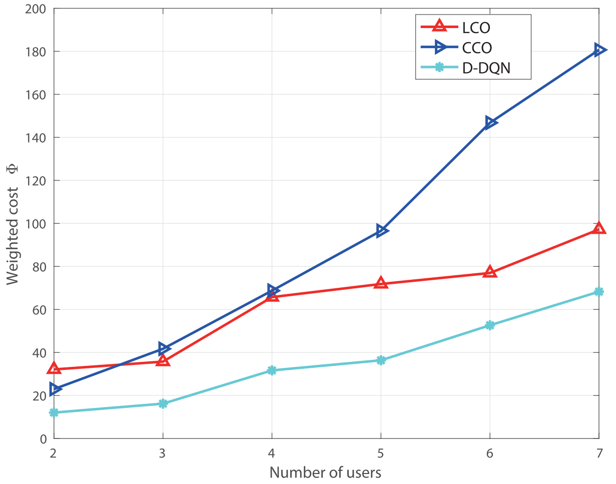

FIGURE 7. Comparison of three offloading strategies with respect to N.

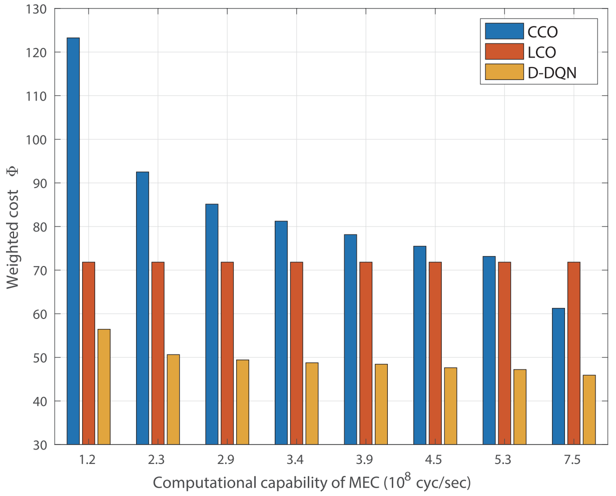

FIGURE 8. The weighted cost with respect to the computational capability at the CAP.

Fig. 6 demonstrates the weighted cost of three offloading strategies with respect to the wireless bandwidth, where W_{total} varies from 2MHz to 10MHz and the computational capability at the CAP is set to 6.3×10^8 cycle/sec. As shown in Fig. 6, we can find that for different values of W_{total}, the proposed scheme achieves a smaller weighted cost compared with the LCO and CCO, which indicates the validity of the proposed scheme. Moreover, the proposed scheme and CCO improve with a larger bandwidth, as the transmission latency and EnC decrease during the offloading process. In further, the performance gap between the proposed scheme and CCO becomes smaller with a larger bandwidth, as the computational tasks tend to be offloaded at the condition of a better wireless channel. Furthermore, the performance of LCO is not affected by the variation of the bandwidth, since the offloading does not occur when the computational tasks are computed at local.

Fig. 7 illustrates the weighted cost of the three offloading strategies with respect to the user number, where W_{total} is 5MHz, the computational capacity at the CAP is set to 6.3×10^8 cycle/sec, and the user number varies from 2 to 7. As shown in Fig. 7, we can see that the system costs of the three offloading strategies all increase with a larger value of N, as an increasing number of users imposes a heavier computational burden on the system. Moreover, the proposed D-DQN based scheme is still better than the conventional LCO and CCO ones, for different value of N, which further shows the validity of the proposed scheme.

Fig. 8 depicts how the system cost of the three offloading strategies varies with the computational capability at the CAP, where W_{total} is 5MHz and there are 5 users in the network. We can observe from Fig. 8 that the proposed scheme achieves a much smaller cost compared to the conventional LCO and CCO ones, for different computational capabilities at the CAP. Moreover, the performances of the proposed and CCO schemes become better with a larger value of computational capability at the CAP, since a more powerful CAP can help reduce the calculation latency that CAP requires. In contrast, LCO has almost the same cost, regardless of

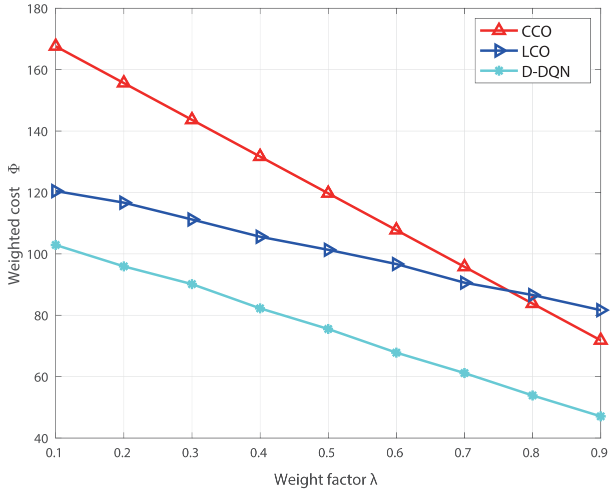

FIGURE 9. **Impact of the weight factor λ on the three offloading strategies.**

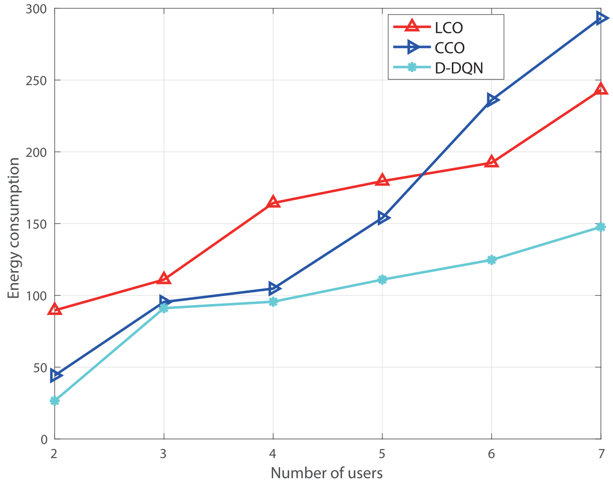

FIGURE 10. **EnC of three offloading strategies with respect to the number of users.**

the computational capability at the CAP. In particular, the performance of LCO is even worse than that of CCO when the computational capability at the CAP is high, as the CAP can help implement the computation efficiently.

Fig. 9 shows how the system cost of the three offloading strategies varies with the weight factor λ, where $N = 5$, the computational capability at the CAP is set to 6.3×10^8 cycle/sec and λ changes in the interval of [0.1,0.9]. We can observe from Fig. 9 that for various weight factors, the proposed scheme achieves a much smaller cost compared with the conventional LCO and CCO ones. Moreover, when $\lambda = 0.9$, CCO has a smaller cost than LCO, as the utilization of the computational resources at the CAP through task offloading is helpful in reducing the EnC. On the other hand, when λ is small, CCO has a much higher cost than LCO, as the transmission latency turns to be the system bottleneck in this case.

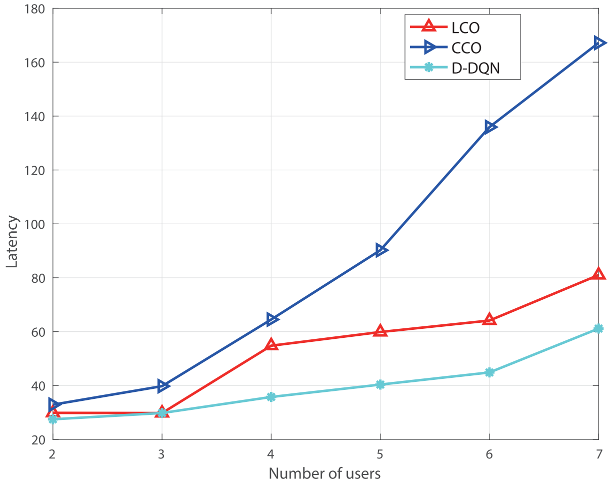

FIGURE 11. **Latency of three offloading strategies with respect to the number of users.**

To further show the advantages of the proposed scheme in the terms of latency and EnC, the weight factor λ is set to 0 and 1, respectively in the experiment. To this aim, Figs. 10 and 11 are plotted to show the EnC and latency of the three offloading strategies with respect to the user number, where W_{total} is 5MHz and the number of users changes from 2 to 7. Specifically, Fig. 10 and Fig. 11 are associated with the EnC and latency, respectively. From these two figures, we can find that the performance gap between the proposed scheme and the conventional ones increases when there are more users. The inherent mechanism is that the proposed scheme is able to exploit the system resources such as communication and computation, and it can achieve a fine trade-off of communication and computation during the offloading process.

V. CONCLUSION

In this paper, we investigated an intelligent MEC network for IoT, where multiple users could offload their confidential tasks to the CAP for the assistance of computation. One UAV eavesdropper existing in the network could overhear the secure data transmission from the users to the CAP. We optimized the system design of the intelligent MEC network, by adaptively allocating the offloading ratio and wireless bandwidth, to reduce the linear combination form of the latency and EnC. Specifically, the deep Q-network (DQN) network was employed to adjust the offloading ratio and transmission width based on the deep reinforcement learning, which could help calculate the computational tasks and suppress the eavesdropping from the UAV efficiently. Simulation results were finally presented to validate the proposed studies. In particular, the proposed offloading strategy could outperform the conventional ones significantly, in the terms of latency and EnC.

The authors of this work would like to point out that one limitation of this work lies in that the instantaneous channel state information of the eavesdropping links should be known, in order to calculate the secrecy data rate. This is however maybe difficult in practice, particularly for the passive eavesdroppers which are hard to detect. In further work, we will estimate the instantaneous channel state information of the eavesdropping links, at least the statistical channel state information of the eavesdropping links, to estimate the secrecy data rate for the considered system. Moreover, we will integrate some other wireless transmission techniques such as the intelligent reflecting surfaces [39], [40] into the considered MEC networks, to further reduce the latency and EnC.

REFERENCES

[1] X. Hu, C. Zhong, Y. Zhu, X. Chen, and Z. Zhang, "Programmable metasurface-based multicast systems: Design and analysis," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1763–1776, Aug. 2020.

[2] K. He, L. He, L. Fan, Y. Deng, G. K. Karagiannidis, and A. Nallanathan, "Learning-based signal detection for MIMO systems with unknown noise statistics," *IEEE Trans. Commun.*, vol. 69, no. 5, pp. 3025–3038, May 2021.

[3] 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.

[4] Y. Guo, Z. Zhao, K. He, S. Lai, J. Xia, and L. Fan, "Efficient and flexible management for industrial Internet of Things: A federated learning approach," *Comput. Netw.*, vol. 192, pp. 1–9, Jun. 2021.

[5] 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.

[6] G. Gui, F. Liu, J. Sun, J. Yang, Z. Zhou, and D. Zhao, "Flight delay prediction based on aviation big data and machine learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 140–150, Jan. 2020.

[7] X. Li, M. Zhao, Y. Liu, L. Li, Z. Ding, and A. Nallanathan, "Secrecy analysis of ambient backscatter NOMA systems under I/Q imbalance," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12286–12290, Oct. 2020.

[8] Z. Pan, X. Yi, Y. Zhang, B. Jeon, and S. Kwong, "Efficient in-loop filtering based on enhanced deep convolutional neural networks for HEVC," *IEEE Trans. Image Process.*, vol. 29, pp. 5352–5366, 2020.

[9] Z. Pan, W. Yu, J. Lei, N. Ling, and S. Kwong, "TSAN: Synthesized view quality enhancement via two-stream attention network for 3D-HEVC," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Feb. 5, 2021, doi: 10.1109/TCSVT.2021.3057518.

[10] J. Zhao, X. Sun, Q. Li, and X. Ma, "Edge caching and computation management for real-time Internet of vehicles: An online and distributed approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 4, pp. 2183–2197, Apr. 2021.

[11] Z. Wang, W. Zhou, L. Chen, F. Zhou, F. Zhu, and L. Fan, "An adaptive deep learning-based UAV receiver design for coded MIMO with correlated noise," *Phys. Commun.*, vol. 47, Aug. 2021, Art. no. 101365.

[12] Z. Pan, F. Yuan, J. Lei, W. Li, N. Ling, and S. Kwong, "MIE-GAN: Mobile image enhancement via a multi-module cascade neural network," *IEEE Trans. Multimedia*, early access, Jan. 28, 2021, doi: 10.1109/TMM.2021.3054509.

[13] X. Li, H. Mengyan, Y. Liu, V. G. Menon, A. Paul, and Z. Ding, "I/Q imbalance aware nonlinear wireless-powered relaying of B5G networks: Security and reliability analysis," *IEEE Trans. Netw. Sci. Eng.*, early access, Sep. 3, 2020, doi: 10.1109/TNSE.2020.3020950.

[14] K. He, Z. Wang, D. Li, F. Zhu, and L. Fan, "Ultra-reliable MU-MIMO detector based on deep learning for 5G/B5G-enabled IoT," *Phys. Commun.*, vol. 43, pp. 1–7, Dec. 2020.

[15] J. Xia, D. Deng, and D. Fan, "A note on implementation methodologies of deep learning-based signal detection for conventional MIMO transmitters," *IEEE Trans. Broadcast.*, vol. 66, no. 3, pp. 744–745, Sep. 2020.

[16] S. Lai, R. Zhao, S. Tang, J. Xia, F. Zhou, and L. Fan, "Intelligent secure mobile edge computing for beyond 5G wireless networks," *Phys. Commun.*, vol. 45, Apr. 2021, Art. no. 101283.

[17] C. Li, J. Xia, F. Liu, D. Li, L. Fan, G. K. Karagiannidis, and A. Nallanathan, "Dynamic offloading for multiuser muti-CAP MEC networks: A deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2922–2927, Mar. 2021.

[18] W. Zhou, "PSO based offloading strategy for cache-enabled mobile edge computing UAV networks," *Cluster Comput.*, pp. 1–8, 2021.

[19] L. Chen, "Intelligent ubiquitous computing for future UAV-enabled MEC network systems," *Cluster Computing*, pp. 1–8, 2021.

[20] S. Tang, W. Zhou, L. Chen, L. Lai, J. Xia, and L. Fan, "Battery-constrained federated edge learning in UAV-enabled IoT for B5G/6G networks," *Phys. Commun.*, vol. 47, Aug. 2021, Art. no. 101381.

[21] R. Zhao, X. Wang, J. Xia, and L. Fan, "Deep reinforcement learning based mobile edge computing for intelligent Internet of Things," *Phys. Commun.*, vol. 43, pp. 1–7, Dec. 2020.

[22] G. Gui, M. Liu, F. Tang, N. Kato, and F. Adachi, "6G: Opening new horizons for integration of comfort, security, and intelligence," *IEEE Wireless Commun.*, vol. 27, no. 5, pp. 126–132, Oct. 2020.

[23] 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.

[24] Y. Zhang, J. Wang, J. Sun, B. Adebisi, H. Gacanin, G. Gui, and F. Adachi, "CV-3DCNN: Complex-valued deep learning for CSI prediction in FDD massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 10, no. 2, pp. 266–270, Feb. 2021.

[25] Y. Wang, G. Gui, T. Ohtsuki, and F. Adachi, "Multi-task learning for generalized automatic modulation classification under non-Gaussian noise with varying SNR conditions," *IEEE Trans. Wireless Commun.*, vol. 20, no. 6, pp. 3587–3596, Jun. 2021, doi: 10.1109/TWC.2021.3052222.

[26] X. Li, Q. Wang, M. Liu, J. Li, H. Peng, M. J. Piran, and L. Li, "Cooperative wireless-powered NOMA relaying for B5G Iot networks with hardware impairments and channel estimation errors," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5453–5467, Apr. 2021.

[27] C. Cheng, L. Guo, T. Wu, J. Sun, G. Gui, B. Adebisi, H. Gacanin, and H. Sari, "Machine learning-aided trajectory prediction and conflict detection for Internet of aerial vehicles," *IEEE Internet Things J.*, early access, Feb. 19, 2021, doi: 10.1109/JIOT.2021.3060904.

[28] X. Li, M. Zhao, M. Zeng, S. Mumtaz, V. G. Menon, Z. Ding, and O. A. Dobre, "Hardware impaired ambient backscatter NOMA systems: Reliability and security," *IEEE Trans. Commun.*, vol. 69, no. 4, pp. 2723–2736, Apr. 2021.

[29] 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.

[30] Z. Zhao, "System optimization of federated learning networks with a constrained latency," *IEEE Trans. Veh. Technol.*, pp. 1–5, 2021.

[31] S. Lai and Y. Guo, "Distributed machine learning for multiuser mobile edge computing systems," *IEEE J. Sel. Topics Signal Process.*, pp. 1–12, 2021.

[32] S. Tang, "Dilated convolution based CSI feedback compression for massive MIMO systems," *IEEE Trans. Veh. Technol.*, pp. 1–5, 2021.

[33] X. Lai, "Cybertwin-driven mobile edge computing for Internet of everything with cochannel interference," *IEEE Trans. Ind. Informat.*, pp. 1–12, 2021.

[34] J. Xia and L. Fan, "Computational intelligence and deep reinforcement learning for next-generation industrial IoT," *IEEE Trans. Netw. Sci. Eng.*, pp. 1–12, 2021.

[35] J. Zhao, J. Liu, L. Yang, B. Ai, and S. Ni, "Future 5G-oriented system for urban rail transit: Opportunities and challenges," *China Commun.*, vol. 18, no. 2, pp. 1–12, Feb. 2021.

[36] X. Lai, "Secure mobile edge computing networks in the presence of multiple eavesdroppers," *IEEE Trans. Commun.*, pp. 1–12, 2021.

[37] 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.

[38] X. Hu, C. Zhong, Y. Zhang, X. Chen, and Z. Zhang, "Location information aided multiple intelligent reflecting surface systems," *IEEE Trans. Commun.*, vol. 68, no. 12, pp. 7948–7962, Dec. 2020.

[39] X. Hu, J. Wang, and C. Zhong, "Statistical CSI based design for intelligent reflecting surface assisted MISO systems," *Sci. China Inf. Sci.*, vol. 63, no. 12, Dec. 2020, Art. no. 222303.

[40] J. Zhang, Y. Zhang, C. Zhong, and Z. Zhang, "Robust design for intelligent reflecting surfaces assisted MISO systems," *IEEE Commun. Lett.*, vol. 24, no. 10, pp. 2353–2357, Oct. 2020.

LIMING CHEN received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 2008, 2010, and 2014, respectively. He is currently a Researcher with the Electrical Power Research Institute of China Southern Power Grid, China. His research interests include power distribution wireless communication, the IoT, cyber security, and smart distribution terminals.

XIAOYUN KUANG (Member, IEEE) received the M.Sc. and M.Sc. degrees in computer science from the South China University of Technology, Guangzhou, China, in 1995 and 1997, respectively. He is currently the Deputy Manager of the Information Security Center, China Southern Power Grid, China. His research interests include computer networks, power distribution wireless communication, and the IoT.

FUSHENG ZHU graduated from the Huazhong University of Science and Technology, in 1996. He received the B.E. degree in electronic and information engineering and the M.B.A. degree from Fudan University, in 2011. He was the Chief Engineer of ZTE Wireless, while he was responsible for research and development of ZTE since he joined the company, in 1996. He is currently working as the President of the Guangdong New Generation Communication and Network Innovative Institute (GDCNi). His research interests include 6G mobile networks, B5G vertical application, and network communication.

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 currently works 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.