EURASIP Journal on Advances in Signal Processing

RESEARCH

Open Access

DQN-based resource allocation for NOMA-MEC-aided multi-source data stream

Jing Ling[1], Junjuan Xia[1*], Fusheng Zhu[2*], Chongzhi Gao[1*], Shiwei Lai[1] and Venki Balasubramanian[3]

*Correspondence: xiajunjuan@gzhu.edu.cn; zhufusheng@gdcni.cn; czgao@gzhu.edu.cn

[1] School of Computer Science, Guangzhou University, Guangzhou, China
[2] Guangdong New Generation Communication and Network Innovative Institute (GDCNi), Guangzhou, China
[3] School of Science, Engineering and Information Technology, Federation University, Mount Helen, VIC 3350, Australia

Abstract

This paper investigates a non-orthogonal multiple access (NOMA)-aided mobile edge computing (MEC) network with multiple sources and one computing access point (CAP), in which NOMA technology is applied to transmit multi-source data streams to CAP for computing. To measure the performance of the considered NOMA-aided MEC network, we first design the system cost as a linear weighting function of energy consumption and delay under the NOMA-aided MEC network. Moreover, we propose a deep Q network (DQN)-based offloading strategy to minimize the system cost by jointly optimizing the offloading ratio and transmission power allocation. Finally, we design experiments to demonstrate the effectiveness of the proposed strategy. Specifically, the designed strategy can decrease the system cost by about 15% compared with local computing when the number of sources is 5.

Keywords: MEC, NOMA, Multi-source data stream, DQN

1 Introduction

In the current society, with the advancement of wireless communication technology [1–4], the quantity of mobile device sources has skyrocketed, which results in the exponential growth of data to be handled [5–7]. However, the local computing capability of the device is often overwhelmed by the huge computing data stream, leading to the slow processing of data streams. To deal with this issue, the cloud server is used to assist devices in computing data stream [8–10], due to its advantages of much computing capacity compared to the local device. However, too many data streams are offloaded to the cloud, which may also bring a serious workload to the cloud server [11]. In addition, the wireless channel is vulnerable, which prolongs the communication delay and affects the system's performance.

Based on the above local computing and cloud server problem, mobile edge computing (MEC) is designed to help to compute data stream [12–14]. In the MEC network, multi-source data streams can be partially offloaded to the computing access point (CAP) to be computed [15]. Because the local device also has computing power, the local and cloud can perform data stream computing at the same time. Therefore, the offloading ratio of data streams becomes a key factor affecting the computing time. The authors in [16] presented an intelligent particle swarm optimization (PSO)-based policy for MEC network unloading based on the cache mechanism, which employed the PSO algorithm to search for a suitable unloading ratio to achieve partial unloading. The PSO algorithm converged quickly and the algorithm was simple, but if the function had multiple local extrema, it was easily trapped in local extrema and cannot be got the optimal solution. The authors of [9, 17] studied a multi-user multi-CAP MEC network with task offloading, where the network environment was time-varying, and the system cost was mainly determined by energy consumption and delay. Besides, a dynamic unloading policy based on DQN was devised. Users could dynamically adjust the unloading ratio to optimize the system cost to ensure the system performance of the MEC network.

Despite the above foundation, the MEC network with dynamical offloading still faces inherent limitations. Limited communication resources are difficult to support the orthogonal multiple access (OMA) of massive users. To deal with this issue, non-orthogonal multiple-access (NOMA), which emerges as a new access technology, can help support the massive users. NOMA is a promising technology for reducing delay and energy consumption in MEC networks. The technology uses non-orthogonal transmission at the transmitter with the allocated source transmission power, introduces scrambling code information, and then removes the scrambling code information at the CAP through successive interference cancellation (SIC) to achieve correct demodulation [18]. Multiple sources share the same bandwidth to send the data stream simultaneously, and the CAP receives the transmitted information and then decodes it. This has a clear advantage in terms of increasing the transmission rate of the data stream [19, 20]. According to this principle, multiple sources can use the same bandwidth to offload data streams to the CAP simultaneously to decrease system energy consumption and delay.

So far, there has been a large number of investigations on the resource allocation of the NOMA-MEC system. For example, the authors in [21] used reinforcement learning to optimize the computation and cache of the multi-server NOMA-MEC system. The author of [22] studied a computing unloading system with the help of NOMA and dual connection (DC) and employed the deep learning-based intelligent unloading method to reduce the total system consumption. The authors in [23] designed a secure communication strategy for the NOMA-assisted UAV-MEC system for large-scale access users. The author of [24] considered a NOMA-MEC network, and jointly optimized the total system energy consumption through the convex theory and the iterative algorithm. Many current studies focus on optimizing the unloading ratio of the MEC network, but when NOMA technology is used to assist the unloading, the transmission power in the unloading stage also has a huge impact on the energy consumption and delay of the system.

On this basis, the author studied multi-source MEC networks where CAPs are deployed at the edge. In the networks, computational data can be offloaded to adjacent CAPs through favorable data stream division and offloading to reach low-energy consumption and low delay [5, 25, 26]. Meanwhile, an optimization strategy based on DQN is proposed for data stream offloading. A deep learning-based Q-learning algorithm integrates neural network techniques and value function approximation [27]. The neural network is trained by using target networks and empirical replay methods. The system cost is designed for a linear weighting function of delay and energy consumption. The offloading decision on MEC networks is modeled as the Markov decision process to employ reinforcement learning methods for resource allocation to improve network performance and reduce system cost [3, 7, 10]. Finally, the designed scheme is verified to be significantly superior through simulation experiments. The significant contributions of the paper are listed in the following:

- We consider a NOMA-aided MEC network with *S* sources and one CAP. Based on this, we propose a linear combination of energy consumption and delay in the system cost to measure the considered network performance. Meanwhile, the offload ratio and transmission power ratio are jointly optimized to lower the cost of the system.
- We come up with a DQN-based data stream unloading optimization policy. In practical applications, the network environment is dynamic, which increases the optimization difficulty. Therefore, we use this strategy to dynamically obtain the allocation of the unloading ratio and transmission power ratio of the system.
- We design experiments to compare different schemes, and the simulation experiment results indicate that the designed DQN-based strategy has a lower total system cost than other methods.

We have organized the rest of the paper as shown below. Following the introduction, we discuss the offloading model of the considered NOMA-aided MEC network in Sect. and give the relevant calculation formula and model optimization formula of the system. After the discussion of the system model, the devised DQN-based method is shown in Sect. . Section presents the results of the simulation experiments. Finally, in Sect. , we conclude the whole work.

2 System model

As Fig. 1 shows, we explore the MEC network with *S* sources and one CAP, where the NOMA technology is applied to assist multi-source data streams for transmission. Data streams are partially processed at local sources with limited computing capability and the other part of data streams are unloaded to CAP to be computed. Considering the

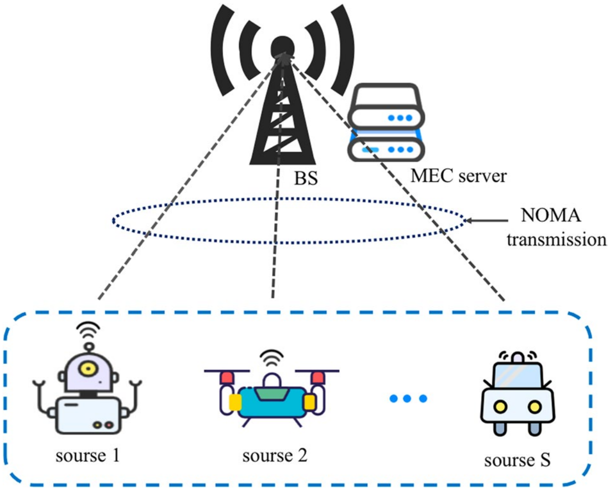

Fig. 1 A NOMA-aided MEC Network with multiple sources and a CAP

performance of the MEC network, we use NOMA to help offload data streams to the CAP with sufficient computing capability through wireless links to accelerate computing. Concretely, the data stream sets of sources in the network are denoted by $\{D_s|1 \leq s \leq S\}$. Each source has different data streams D_s which has q_s number of bits and offloads a part of the data stream to the server to be computed. After calculating the offloaded data stream, the CAP returns the results to the source via dedicated feedback links. The following sections present the data stream offloading model, the local computing model, and the CAP computing model, respectively, specified in the following.

2.1 Data stream offloading model

In this part, we describe the data stream offloading model. When part of the data stream is offloaded to the CAP to be computed, multiple sources need to transmit the offloaded stream over the radio link using NOMA. The transmission rate of source D_s can be described as

$$r_s = B\log_2(1 + \tfrac{P_s|h_s|^2}{\sum_{n=1}^{s-1} P_n|h_n|^2+\sigma^2}), \tag{1}$$

where B is the bandwidth of wireless channel from D_s to CAP, P_s is the transmission power of source D_s, and $|h_s|^2$ is the channel gain of wireless channel from the source D_s to CAP. The symbol σ^2 stands for the noise of AWGN [28–31]. As previously stated, we assume $P_1|h_1|^2 \leq P_2|h_2|^2 \leq \cdots \leq P_s|h_s|^2$.

At each time slot, sources D_s have q_s bits of the data stream that need to be processed and offload a part of the data stream to CAP through the wireless link. The transmission delay of the offloaded data stream of sources D_s is [17, 32]

$$t_s = \frac{\beta_s q_s}{r_s}, \tag{2}$$

where $\boldsymbol{\beta_s} = [\beta_1, \beta_2, \ldots, \beta_s]$ represents the percentage of source D_s to be unloaded to server which satisfies $\beta_s \in [0, 1]$. Since multi-source data streams are unloaded in parallel, the total delay in the unloading phase is

$$T_1 = \max\{t_1, t_2, \ldots, t_S\}. \tag{3}$$

In addition, the system energy consumption in the unloading phase can be obtained by

$$E_1 = \sum_{s=1}^{S} t_s P_s. \tag{4}$$

2.2 Local computing model

As mentioned above, some multi-source data streams can be calculated locally. The local calculation delay of source D_s can express as

$$t^s_{\text{local}} = \frac{(1-\beta_s)c_s}{f_s}, \tag{5}$$

where f_s refers to local computing capability on source D_s and c_s represents the CPU cycle required for processing multi-source data stream on source D_s. The total local calculated time is

$$T_2 = \max\left\{t_{\text{local}}^1, t_{\text{local}}^2, \ldots, t_{\text{local}}^S\right\}. \tag{6}$$

The total local calculated energy consumption is

$$E_2 = \sum_{s=1}^{S} t_{\text{local}}^s P_{\text{local}}^s. \tag{7}$$

2.3 CAP computing model

After part of the data stream on source D_s is successfully unloaded to the CAP through wireless links, the offloaded data stream will be computed at CAP. The computation delay at the CAP of source D_s is

$$t_{\text{MEC}}^s = \frac{\beta_s c_s}{F_s}, \tag{8}$$

where F_s denotes the computational power allocated to each source of the CAP. Different offloaded data streams are calculated in parallel at the CAP, so the total calculation delay at the CAP is

$$T_3 = \max\left\{t_{\text{MEC}}^1, t_{\text{MEC}}^2, \ldots, t_{\text{MEC}}^s\right\}. \tag{9}$$

Meanwhile, the energy consumption produced at the CAP is

$$E_3 = \sum_{s=1}^{S} t_{\text{MEC}}^s P_{\text{MEC}}^s. \tag{10}$$

Since local operation and unloading can be performed at the same time, the total calculation delay of multi-source data streams is

$$T_{\text{total}} = \max\{T_2, T_1 + T_3\}. \tag{11}$$

Moreover, the total energy consumption equation is

$$E_{\text{total}} = E_1 + E_2 + E_3. \tag{12}$$

The total system energy consumption is the sum of energy consumption produced at each stage which is related to the delay at this stage. However, it is hard to measure the system behavior only by the total system energy consumption, because the total delay is the maximum of the local delay and the sum of delays produced at other stages. To comprehensively measure system performance, we devise the total system cost as a linear weighted function of total delay and total energy consumption [33], which can be described as

$$\Theta_s = \mu T_{\text{total}} + (1-\mu)E_{\text{total}}, \tag{13}$$

where $\mu \in$ [0,1] is a weight factor. Notice that, when $\mu = 0$, the system cost consists directly of the system energy consumption, and we focus on the impact of system energy consumption on the considered system. When $\mu = 1$, the system cost consists only of the system delay, and we pay attention to the impact of delay on the considered MEC network. In the considered system, we can change the value of μ to meet the requirements of different scenarios on energy consumption and time.

2.4 Problem formulation

As mentioned before, the system cost can be used to measure the system's performance. To improve the system performance, the data stream needed to be processed with the minimum system cost. Therefore, we formulate this problem as minimizing the system cost by optimizing the unloading data stream and transmission power allocation, which can be expressed as

$$\begin{aligned} &\min_{\{\beta_s,\alpha_s\}} \Theta_s \\ &\quad \text{s.t. } C_1 : \beta_s \in [0,1], \forall s \in [1,S], \\ &\qquad\;\; C_2 : \alpha_s \in (0,1], P_s = \alpha_s P_{\max}^s, \end{aligned} \tag{14}$$

where C_1 is a constraint on the unloading ratio, representing the limitation of the portion of the data stream unloaded to the CAP. Constraint C_2 proposes the constraint of transmission power allocation, where $P_{\max}^s$ represents the maximum transmission power. The α_s is the transmission power allocation ratio between the sth source and the CAP. Due to the complexity of this problem, we employ DQN to solve this problem which is introduced in the next section. We summarize the notations mentioned in this section in Table 1.

3 DQN-based offloading policy

The unloading policy determines the part of the data stream unloaded to the CAP and the transmission power allocation on the CAP, which significantly affects the performance of the system. In this section, we first formulate the optimization problem as a Markov decision process(MDP). Then we investigate a DQN-based optimization scheme to obtain the offload ratio and transmission power ratio to minimize the system cost.

In MDP, at time slot τ, the state of the environment is s_τ. The agents first obtain the action a_τ from the policy π_τ according to s_τ. Then the agents perform action a_τ at the environment, resulting in the environment shift from s_τ to $s_{\tau+1}$ and obtaining the reward r_τ from the environment. Concretely, the state space is

$$\boldsymbol{s} = \{\boldsymbol{\beta}, \boldsymbol{\alpha}\}, \tag{15}$$

where $\boldsymbol{\beta} = \{\beta_1, \beta_2, \beta_3, \ldots, \beta_s\}$ is the unloading ratio of the multi-source data stream, $\boldsymbol{\alpha} = \{\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_s\}$ is the transmission power allocation ratio at CAP. Besides, the action space is $\boldsymbol{A} = \left\{\delta_1, \delta_1^*, \delta_2, \delta_2^*, \delta_3, \delta_3^*, \ldots, \delta_s, \delta_s^*, \varrho_1, \varrho_1^*, \varrho_2, \varrho_2^*, \varrho_3, \varrho_3^*, \ldots, \varrho_s, \varrho_s^*\right\}$, where $\delta_s = -\theta$ and $\delta_s^* = +\theta$ are the acts of adjusting the unloading ratio under the constraints C_1, and $\varrho_s = -\theta$ and $\varrho_s^* = +\theta$ are the actions to adapt the transmit power allocation ratio with the constraint C_2. For minimizing the system cost of the considered NOMA-aided MEC system, the reward is designed as

Table 1 Symbol notations

Notation	Definition
S	Number of sources
D_s	The sth sources
q_s	The bits of the sth source data stream
c_s	The CPU cycles of the sth source data stream
β_s	The percentage of source D_s data stream to be unloaded to the CAP
$\|h_s\|^2$	Channel gain of wireless channel from the source D_s offloaded to the CAP
P_s	Transmit power of source D_s
B	Bandwidth of wireless channel from D_s to CAP
t_s	Calculation delay of the sth source data stream unloading to the CAP
T_1	The delay in the unloading phase
E_1	System energy consumption in the unloading phase
f_s	local computing capability on source D_s
t^s_{local}	Local calculation delay of the sth source
T_2	Total local calculated time
P^s_{local}	Local calculated power of the sth source
E_2	Total total local calculated energy consumption
F_s	The computational power allocated to each source of the CAP
t^s_{MEC}	Computation delay at the CAP of source D_s
T_3	Total calculation delay at the CAP
P^s_{MEC}	The calculated power of the CAP
E_3	System energy consumption produced at the CAP
T_{total}	Total calculation delay of multi-source data streams
E_{total}	Total energy consumption
P^s_{max}	Maximum transmit power of the source
μ	A weight factor of system cost
α_s	Transmit power ratio of source D_s
Θ_s	System cost

$$r_\tau = \begin{cases} -\gamma_1 \;\; If \;\; \Theta(\tau) > \Theta(\tau+1), \\ -\gamma_2 \;\; If \;\; \Theta(\tau) = \Theta(\tau+1), \\ \gamma_1 \quad If \;\; \Theta(\tau) < \Theta(\tau+1), \end{cases} \tag{16}$$

where $\gamma_1 > \gamma_2 > 0$. Notice that, if the execution of a_τ results in the reduction of system cost of the environment in time slot $\tau + 1$, the agents obtain a positive reward. On the contrary, the reward is negative. Moreover, the Q function which measures the performance of the action in the current environment is used in obtaining the best policy. It can be expressed as

$$\pi^* = \arg\max_\pi Q_\pi(s, a). \tag{17}$$

As mentioned above, the optimization problem of the considered NOMA-aided MEC network can be modeled as MDP. Therefore, we employ a deep learning method, DQN, to solve this problem.

As shown in Fig. 2, the DQN consists of two networks and one replay memory. The replay memory is used to store transition samples $(s_\tau, a_\tau, r_\tau, s_{\tau+1})$. The evaluation

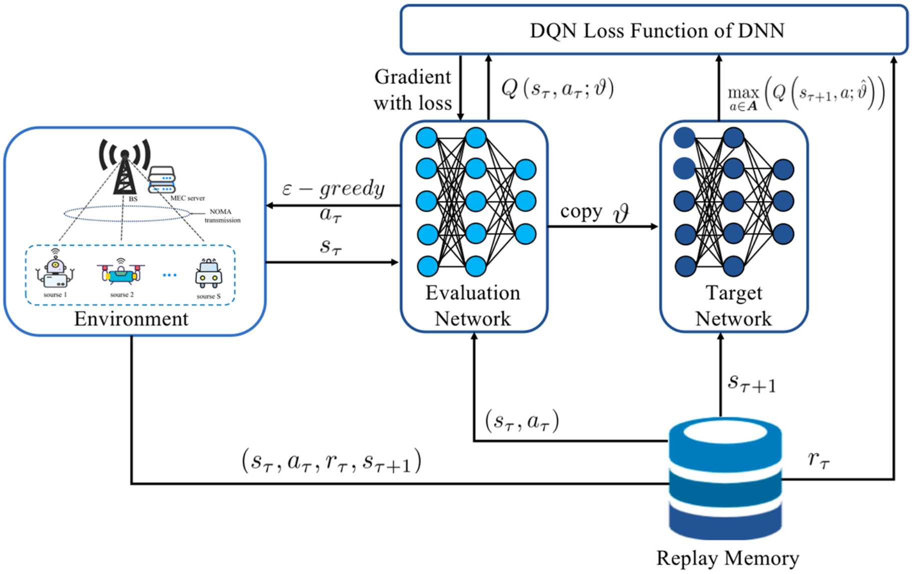

Fig. 2 DQN-based offload policy framework

network outputs the action $a_\tau \in \boldsymbol{A}$ with the input state $s_\tau \in \boldsymbol{s}$. When the replay memory has enough data, the evaluation network begins training and updates weights ϑ every step. The target network, which helps train the evaluation network, is initialed as an evaluation network at the beginning and updated as the evaluation network at every certain step.

To avoid the network falling into local optimization, ε-greedy strategy is used to help the agents to explore. It can be expressed as

$$a_\tau = \begin{cases} \arg\max_{a\in \boldsymbol{A}} Q(s_\tau, a; \vartheta), & \text{with probability } 1-\epsilon, \\ \text{randomly choose}, & \text{otherwise}, \end{cases} \tag{18}$$

where ϑ is the weight of the evaluation network. We employ the temporal difference (TD) approach to support training DQN by defining the TD-target obtained by the target network [33].

$$Q(s_\tau, a_\tau; \vartheta) = r_\tau + \varphi \max_{a\in \boldsymbol{A}} (Q(s_{\tau+1}, a; \vartheta)), \tag{19}$$

where φ is weight factor.

Moreover, we give the loss function [2, 34] based on TD-target as

$$L_\tau = \left(\left(r_\tau + \varphi \max_{a\in \boldsymbol{A}}\left(Q\left(s_{\tau+1}, a; \hat\vartheta\right)\right)\right) - Q(s_\tau, a_\tau; \vartheta)\right)^2. \tag{20}$$

Based on the above, we summarize the DQN-based unloading policy in Algorithm 1

Algorithm 1 DQN based unloading policy

```
Input:
    Maximum episodes $eps$;
    Maximum iterations $i_{max}$;
Output:
    $\boldsymbol{\beta}$, $\boldsymbol{\alpha}$;
Initialize experience replay $R$;
Initialize the weights $\vartheta$ of the evaluation network and the weights $\hat{\vartheta}$ of the target network randomly,
    set $\hat{\vartheta} = \vartheta$;
for each episodes $e = 1 : eps$ do
    Initialize $s_\tau \in \boldsymbol{s}$
    Initialize $|h_s|^2$;
    for each iterations $i = 1 : i_{max}$ do
        Select $a_\tau \in \boldsymbol{A}$;
        Calculate the unloading ratio $\boldsymbol{\beta}$ and transmission power ratio $\boldsymbol{\alpha}$ through $a_\tau$;
        Calculate the total cost $\Theta_s$ through $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ to obtain the reward $r_\tau$ and the status $s_{\tau+1}$;
        Store the transition sample $(s_\tau, a_\tau, r_\tau, s_{\tau+1})$ in $R$;
        Sample a minibatch of transitions from $R$ randomly;
        Update $\vartheta$;
        Every time slots reset $\hat{\vartheta} = \vartheta$;
        Set $s_\tau = s_{\tau+1}$;
    end for
end for
```

4 Results and discussion

This section demonstrates the advantage of the designed DQN-based unloading strategy in the considered NOMA-aided MEC network through simulations. In the considered MEC network, with all channels experiencing Rayleigh flat fading [35–37]. We set the number of sources to 5 in our experiment. The size of multi-source data streams is set to 14 Mb, 3 Mb, 16 Mb, 8 Mb, and 18 Mb, and the computing power of each source is set to $5 \times (10^7)$ cycle/s. Besides, the maximum transmission power and calculating power of each source are 1 W and 1.5 W. The computing power allocated to each source at the CAP is $8 \times (10^7)$ cycle/s, and the calculating power is 1.5 W. In addition, the total bandwidth is set to 6 MHz. The detailed network parameter settings are shown in Table 2.

Table 2 Parameter setting

Notation	Definition
The number of source S	5
The variance of the AWGN	0.01
Wireless bandwidth B	6 MHz
Local computing capability f_s	$5 \times (10^7)$ cycle/s
Maximum transmit power P^s_{max}	1W
Local calculated power of the sth source P^s_{local}	1.5W
The calculated power of the CAP P^s_{MEC}	1.5W
The computing power allocated to each source of the CAP F_s	$8 \times (10^7)$ cycle/s
The weight factor of system cost μ	0.5
The loss function of neural network in DQN	10^{-4}
Learning rates of neural network in DQN	10^{-2}
The capability of experience replay	$2 \times (10^4)$
The batch size of neural network in DQN	64

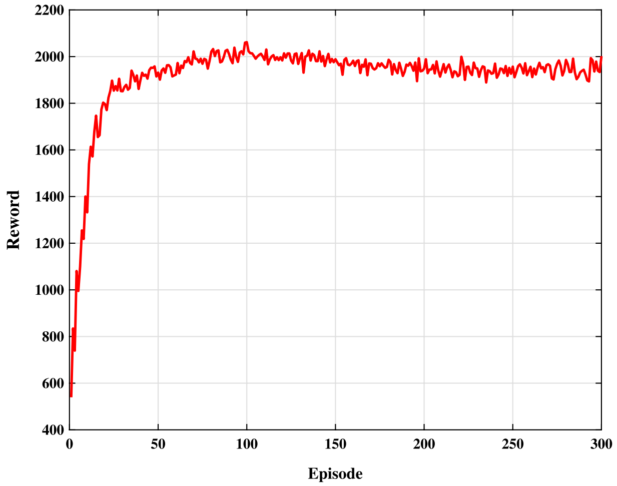

Fig. 3 The total reward of each episode during training of designed policy

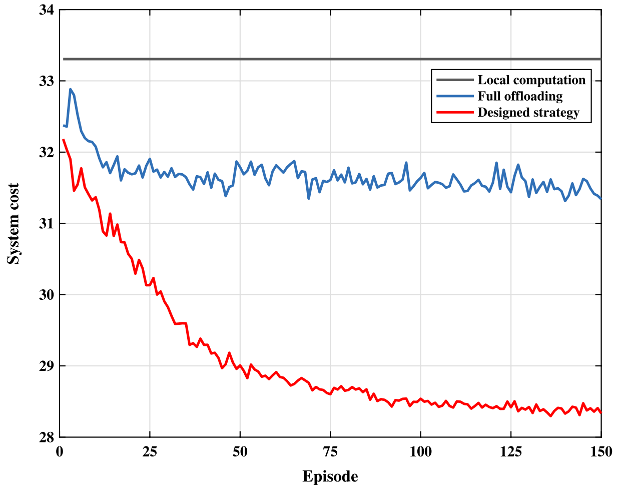

Fig. 4 The convergence of the designed strategy versus episode

Figure 3 shows the reward of each episode during the training of the proposed DQN-based offloading policy, where $\mu = 0.5$ and the DQN has trained 300 episodes at all. From this figure, we can see that the reward grows rapidly in the previous 20 episodes. After about 50 episodes, the reward fluctuates slightly around 1950. It shows that our training converges, and helps to verify the efficacy of the designed DQN-based policy.

Figure 4 plots system cost versus the number of training episodes under the three scenarios, where the number of sources is set to 5 and we train 150 episodes at all. For comparison, we also plot the cost of the other two scenarios. One is the local computation scenario, where each source data stream is computed locally; the other is the full offloading scenario, where data streams on each source are all offloaded to CAP for computing and the computing capability allocation on CAP is obtained by DQN. From this plot, we could notice that the system cost of the designed strategy declines sharply during the previous 100 episodes and converges at about 28.30 after 100 episodes. The system cost of the full offloading scenario decreases gradually during the previous 20 episodes and converges at about 31.6 after 20 episodes. On the contrary, the system cost of the local computation scenarios remains at 33.30 during the whole training. This result illustrates that the designed strategy has the best performance in reducing the system cost among the three proposed solutions. And the designed strategy can give a good unloading and resource allocation strategy for the considered NOMA-aided MEC network.

Figure 5 shows the relationship between the weighting factor μ and the system cost Θ, where μ varies from 0.1 to 0.9 and the number of sources is set to 5. We can see from Fig. 5 that the system cost obtained by the designed strategy is lower than that of the local computation and full offloading schemes with different values of μ. This indicates that the proposed scheme can significantly and efficiently improve the performance of the considered NOMA-aided MEC network by reasonably allocating resources and unloading rate. Moreover, the total system cost of these three schemes decreases when μ varies from 0.1 to 0.9. The main reason for the reduction of system cost is that the system cost is a linearly weighted function of time and energy consumption, where the impact of energy consumption is greater than the delay. The increase in μ magnifies the impact of energy consumption, thus making the total system cost significantly lower.

Figure 6 presents the influence of wireless bandwidth B on system cost, where $S = 5$ and the value of bandwidth varies from 5 to 9 MHz. As shown in Fig. 6, the system cost of the designed strategy is lower than that of other schemes under the different values of bandwidth. This result indicates that our designed strategy outperforms other schemes under different communication environments. Moreover, the system cost of the designed scheme and the full offloading decrease, and the system cost of the local computation solutions stays at the same value as bandwidths increase. The reason is that the bigger bandwidth reduces the unloading cost of the designed policy and the full offloading scheme, while the data stream is calculated locally in the local computation scheme resulting in non-unloading cost.

Figure 7 presents the impact on the system cost Θ of the variation of the computing power allocated to each source of the CAP in three scenarios, where the number

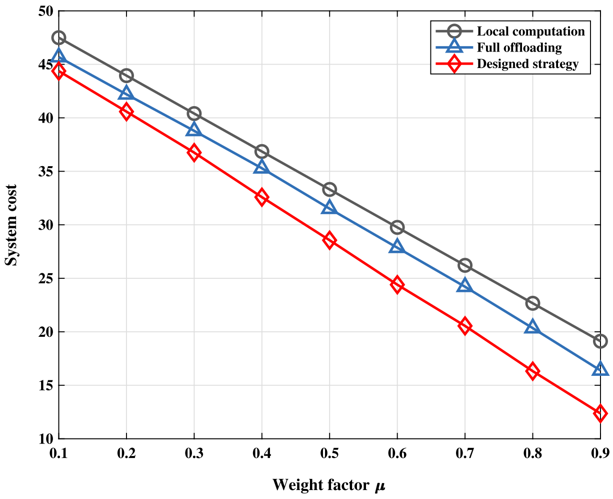

Fig. 5 Impact of the weight factor μ on the system cost

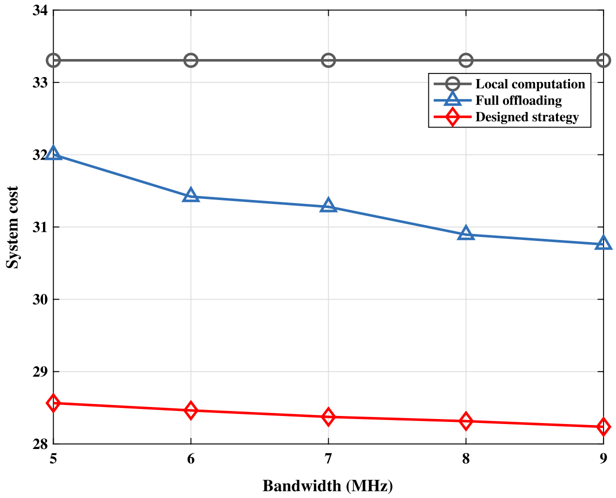

Fig. 6 System cost comparison of three schemes under the different values of bandwidth

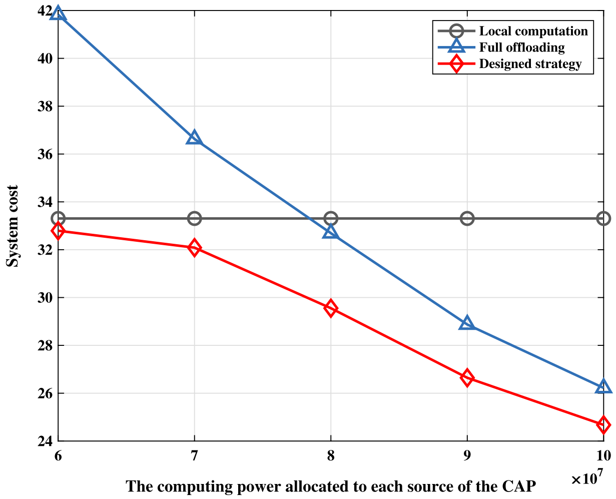

Fig. 7 The influence of the computing power allocated to each source of the CAP on system cost

of sources is 5 and the computing power allocated to each source at the CAP changes from 6×10^7 cycle/s to 10×10^7 cycle/s. As shown in this figure, the system cost of the designed strategy and full offloading gradually decreases when the computing power allocated to each source at the CAP increases from 6×10^7 cycle/s to 10×10^7 cycle/s. The reason is that the CAP with more computational power can compute data streams faster, resulting in reduced system energy consumption and delay. Moreover, the system cost of the full offloading tends to decrease faster than that of the designed strategy. This is because the designed strategy is more robust in the various MEC environment and the full offloading solution is more impacted by the computational power at the CAP. This implies that the designed strategy obviously superior to the full offloading and local computation schemes.

Figure 8 plots the system cost of the designed strategy versus the scale of the MEC network, where S changes from 2 to 6 and the local computing capability is set to 5×10^7

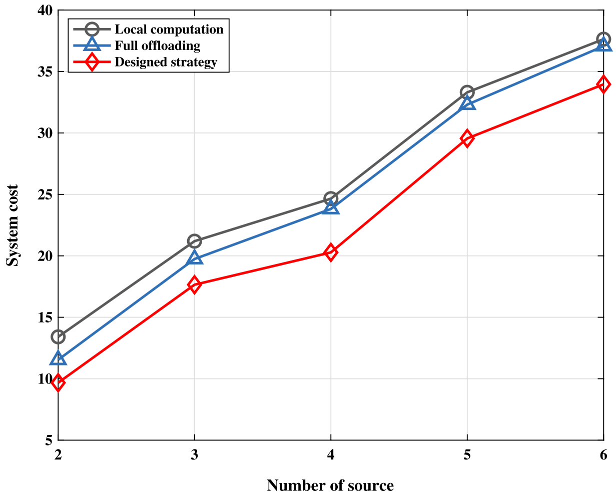

Fig. 8 The influence of the number of sources on the system cost

cycle/s. From this plot, we can notice that the system cost of the designed strategy is always lower than that of local computing schemes and full offloading schemes as the value of S rises. It indicates that the designed strategy can improve the performance of considered NOMA-aided MEC networks with different scales. Moreover, the system cost of the three schemes rises as the value of S varies from 2 to 6. This is due to the increasing number of sources leading to the computing data streams increasing for local and CAP. It leads to system delay grows.

5 Conclusion

This paper has investigated a NOMA-aided MEC network with multi-source and one CAP, in which multi-source data streams were partially offloaded to CAP to accelerate computing. In the considered NOMA-aided MEC network, we designed the system cost as the linear weighting function of energy consumption and delay produced in the unloading process. For reducing the cost of the considered network, we proposed the DQN-based offloading strategy for minimizing the system cost by optimizing the transmission power ratio and the offloading ratio during the unloading process. We compared the proposed DQN-based offloading strategy and other methods through experiments. The experimental results showed that the proposed method was more effective than other methods under different communication environments with various bandwidths. Moreover, in different NOMA-aided MEC networks with different scales, different transmission capabilities, different bandwidth, or different computing capabilities, the proposed DQN-based offloading strategy is more robust than other methods with minimum system cost. Specifically, the designed strategy has been able to decrease the system cost by about 15% compared with local computing when the number of sources is 5.

Abbreviations

MEC	Mobile edge computing
NOMA	Non-orthogonal multiple access
DQN	Deep Q network
CAP	Computing access point
PSO	Particle swarm optimization

SIC	Successive interference cancellation
DC	Dual connection
UAV	Unmanned aerial vehicle
MDP	Markov decision process
TD	Temporal difference
AWGN	Additive white Gaussian noise

Acknowledgements
None.

Author contributions
JL designed the proposed framework and conducted the simulations, JX assisted in revising the manuscript for structure and grammar checking, VB helped to perfect the optimization method, F. Zhu assisted to optimize the design of the deep neural network, CG aided in conducting the simulations in this work, and SL helped to interpret the simulation results in this work. JX, FZ, and CG are the corresponding authors of this paper. All authors read and approved the final manuscript.

Funding
This work was supported by the Key-Area Research and Development Program of Guangdong Province, China (No. 2019B090904014), Science and Technology Projects in Guangzhou (No. 202102010412), and Yangcheng Scholars Research Project of Guangzhou (No. 202032832), and by Science and Technology Program of Guangzhou (No. 202201010047).

Availability and data materials
The authors state the data available in this manuscript.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing Interests
The authors declare that they have no competing interests.

Received: 3 January 2023 Accepted: 24 March 2023
Published online: 20 April 2023

References

1. W. Wu, F. Zhou, R.Q. Hu, B. Wang, Energy-efficient resource allocation for secure noma-enabled mobile edge computing networks. IEEE Trans. Commun. **68**(1), 493–505 (2020)
2. L. Chen, X. Lei, Relay-assisted federated edge learning: performance analysis and system optimization. IEEE Trans. Commun. **PP**(99), 1–12 (2022)
3. R. Zhao, M. Tang, Profit maximization in cache-aided intelligent computing networks. Phys. Commun. **PP**(99), 1–10 (2022)
4. J. Ren, X. Lei, Z. Peng, X. Tang, O.A. Dobre, Ris-assisted cooperative NOMA with SWIPT. IEEE Wirel. Commun. Lett. (2023)
5. X. Liu, C. Sun, M. Zhou, C. Wu, B. Peng, P. Li, Reinforcement learning-based multislot double-threshold spectrum sensing with Bayesian fusion for industrial big spectrum data. IEEE Trans. Ind. Inform. **17**(5), 3391–3400 (2021)
6. Z. Na, B. Li, X. Liu, J. Wan, M. Zhang, Y. Liu, B. Mao, Uav-based wide-area internet of things: An integrated deployment architecture. IEEE Netw. **35**(5), 122–128 (2021)
7. W. Zhou, F. Zhou, Profit maximization for cache-enabled vehicular mobile edge computing networks. IEEE Trans. Veh. Technol. **PP**(99), 1–6 (2023)
8. W. Xu, Z. Yang, D.W.K. Ng, M. Levorato, Y.C. Eldar, M. Debbah, Edge learning for B5G networks with distributed signal processing: Semantic communication, edge computing, and wireless sensing. IEEE J. Sel. Top. Signal Process. arXiv:2206.00422 (2023)
9. X. Zheng, C. Gao, Intelligent computing for WPT-MEC aided multi-source data stream. to appear in EURASIP J. Adv. Signal Process. **2023**(1) (2023)
10. S. Tang, L. Chen, Computational intelligence and deep learning for next-generation edge-enabled industrial IoT. IEEE Trans. Netw. Sci. Eng. **9**(3), 105–117 (2022)
11. W. Wu, F. Zhou, B. Wang, Q. Wu, C. Dong, R.Q. Hu, Unmanned aerial vehicle swarm-enabled edge computing: potentials, promising technologies, and challenges. IEEE Wirel. Commun. **29**(4), 78–85 (2022)
12. W. Zhou, X. Lei, Priority-aware resource scheduling for uav-mounted mobile edge computing networks. IEEE Trans. Veh. Technol. **PP**(99), 1–6 (2023)
13. L. Zhang, C. Gao, Deep reinforcement learning based IRS-assisted mobile edge computing under physical-layer security. Phys. Commun. **55**, 101896 (2022)

14. L. Chen, Physical-layer security on mobile edge computing for emerging cyber physical systems. Comput. Commun. **194**(1), 180–188 (2022)
15. Y. Wu, C. Gao, Task offloading for vehicular edge computing with imperfect CSI: a deep reinforcement approach. Phys. Commun. **55**, 101867 (2022)
16. W. Zhou, L. Chen, S. Tang, L. Lai, J. Xia, F. Zhou, L. Fan, Offloading strategy with PSO for mobile edge computing based on cache mechanism. Clust. Comput. **25**(4), 2389–2401 (2022)
17. R. Zhao, C. Fan, J. Ou, D. Fan, J. Ou, M. Tang, Impact of direct links on intelligent reflect surface-aided mec networks. Phys. Commun. **55**, 101905 (2022)
18. Z. Ding, D.W.K. Ng, R. Schober, H.V. Poor, Delay minimization for NOMA-MEC offloading. IEEE Signal Process. Lett. **25**(12), 1875–1879 (2018)
19. X. Liu, Q. Sun, W. Lu, C. Wu, H. Ding, Big-data-based intelligent spectrum sensing for heterogeneous spectrum communications in 5g. IEEE Wirel. Commun. **27**(5), 67–73 (2020)
20. Z. Na, Y. Liu, J. Shi, C. Liu, Z. Gao, Uav-supported clustered NOMA for 6g-enabled internet of things: Trajectory planning and resource allocation. IEEE Internet Things J. **8**(20), 15041–15048 (2021)
21. S. Li, B. Li, W. Zhao, Joint optimization of caching and computation in multi-server NOMA-MEC system via reinforcement learning. IEEE Access **8**, 112762–112771 (2020)
22. C. Li, H. Wang, R. Song, Intelligent offloading for noma-assisted MEC via dual connectivity. IEEE Internet Things J. **8**(4), 2802–2813 (2021)
23. W. Lu, Y. Ding, Y. Gao, Y. Chen, N. Zhao, Z. Ding, A. Nallanathan, Secure noma-based UAV-MEC network towards a flying eavesdropper. IEEE Trans. Commun. **70**(5), 3364–3376 (2022)
24. L. Shi, Y. Ye, X. Chu, G. Lu, Computation energy efficiency maximization for a noma-based WPT-MEC network. IEEE Internet Things J. **8**(13), 10731–10744 (2021)
25. X. Liu, H. Ding, S. Hu, Uplink resource allocation for noma-based hybrid spectrum access in 6g-enabled cognitive internet of things. IEEE Internet Things J. **8**(20), 15049–15058 (2021)
26. X. Liu, C. Sun, W. Yu, M. Zhou, Reinforcement-learning-based dynamic spectrum access for software-defined cognitive industrial internet of things. IEEE Trans. Ind. Inform. **18**(6), 4244–4253 (2022)
27. B. Li, Z. Fei, J. Shen, X. Jiang, X. Zhong, Dynamic offloading for energy harvesting mobile edge computing: architecture, case studies, and future directions. IEEE Access **7**, 79877–79886 (2019)
28. L. He, X. Tang, Learning-based MIMO detection with dynamic spatial modulation. IEEE Trans. Cogn. Commun. Netw **PP**(99), 1–12 (2023)
29. L. Zhang, S. Tang, Scoring Aided Federated Learning on Long-tailed Data for Wireless IoMT based Healthcare System. IEEE J. Biomed. Health Inform. **PP**(99), 1–12 (2023)
30. J. Li, S. Dang, Y. Huang, Composite multiple-mode orthogonal frequency division multiplexing with index modulation. IEEE Trans. Wirel. Commun. (2023)
31. S. Tang, X. Lei, Collaborative cache-aided relaying networks: performance evaluation and system optimization. IEEE J. Sel. Areas Commun. **41**(3), 706–719 (2023)
32. J. Lu, M. Tang, Performance analysis for IRS-assisted MEC networks with unit selection. Phys. Commun. **55**, 101869 (2022)
33. C. Li, J. Xia, F. Liu, D. Li, L. Fan, G.K. Karagiannidis, A. Nallanathan, Dynamic offloading for multiuser muti-cap MEC networks: A deep reinforcement learning approach. IEEE Trans. Veh. Technol. **70**(3), 2922–2927 (2021)
34. Y. Wu, C. Gao, Intelligent resource allocation scheme for cloud-edge-end framework aided multi-source data stream. EURASIP J. Adv. Signal Process. **2023**(1) (2023, to appear)
35. W. Zhou, C. Li, M. Hua, Worst-case robust MIMO transmission based on subgradient projection. IEEE Commun. Lett. **25**(1), 239–243 (2021)
36. J. Li, S. Dang, M. Wen, Index modulation multiple access for 6G communications: principles, applications, and challenges. IEEE Netw. (2023)
37. S. Tang, Dilated convolution based CSI feedback compression for massive MIMO systems. IEEE Trans. Veh. Technol. **71**(5), 211–216 (2022)

Publisher's Note

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