Hindawi
Security and Communication Networks
Volume 2022, Article ID 8072493, 11 pages
https://doi.org/10.1155/2022/8072493

Research Article

Dynamic Resource Allocation and Task Offloading for NOMA-Enabled IoT Services in MEC

Hua Xing,[1] **Jiajie Xu**,[1] **Jintao Hu**,[1] **Ying Chen**,[1] **and Jiwei Huang**[2]

[1]*School of Computer Science, Beijing Information Science and Technology University, Beijing 100101, China*
[2]*Beijing Key Laboratory of Petroleum Data Mining, China University of Petroleum, Beijing 102249, China*

Correspondence should be addressed to Ying Chen; chenying@bistu.edu.cn

Received 31 May 2022; Revised 13 July 2022; Accepted 20 July 2022; Published 31 August 2022

Academic Editor: Xiaolong Xu

Integrating nonorthogonal multiple access (NOMA) and edge computing into the Internet of Things (IoT) for resource allocation and computing offloading can effectively reduce delay and energy consumption and improve spectrum efficiency. Computation tasks can be split into several independent subtasks and can be locally processed by IoT devices or offloaded to the MEC servers to process. The limited computing resources deteriorate the system performance. Thus, it is crucial to design the reasonable allocation strategies of computation resource and transmission power resource. In this paper, we jointly determine the CPU-cycle frequency allocation and transmission power allocation and formulate a stochastic optimization to minimize the energy consumption of IoT devices. Based on the Lyapunov optimization theory, we decompose the optimization problem into two deterministic subproblems to solve separately. One of them is obtained by seeking the first derivative, and the other is solved by using the best response idea after establishing the game model. Meanwhile, we propose a dynamic resource allocation and task offloading (DRATO) algorithm. Moreover, the simulation experiments show that the proposed algorithm effectively improves system performance and reduces energy consumption compared to three other benchmark methods.

1. Introduction

With the rapid development of Internet of Things (IoT) and mobile communication technology, there are more and more types of IoT devices and IoT applications, which provide abundant and convenient services for end users [1–3]. However, the computing load and energy load for IoT devices are also aggravated [4–6]. The computing resource and energy resource of IoT devices are limited, which cannot effectively support these applications [7, 8]. The mobile cloud computing (MCC) allows IoT devices to offload computing tasks to the cloud servers for processing and make use of the powerful computing power to make up for the shortage of computing resources [9–11]. However, the long-distance transmission of tasks will lead to high transmission delay and communication energy consumption.

The mobile edge computing (MEC) technology is proposed as a new computing paradigm to tackle this challenge [12–14]. It can provide computing resources and service to edge devices and users. Specifically, the IoT devices can offload local tasks to the MEC server deployed in the BS for processing, which can effectively reduce system delay and energy consumption [15–17]. With the connection demand of massive devices and increasingly scarce spectrum resources, the traditional orthogonal multiple access (OMA) technology cannot meet the communication demand. Thus, the nonorthogonal multiple access (NOMA) has attracted extensive attention as a new generation of mobile communication access technology. The NOMA technology enables multiple devices to be multiplexed on the same time-frequency resource block, which increases the number of user access [18]. In the uplink, BS can use successive interference cancellation (SIC) technology to effectively eliminate the interference among IoT devices, which can improve the spectrum resource utilization.

In particular, the current research works focus on the application of NOMA in the single BS and multiuser scenario, which only studies the intracell interference among devices and ignores the intercell interference [19]. Based on the above analysis, our paper considers the resource allocation and computing offloading scenario based NOMA among multiple BSs and multiple devices. We aim to minimize the energy consumption of IoT devices. According to the Lyapunov optimization technology, the optimization problem is divided into two parts and solved separately. This includes CPU-cycle frequency allocation and transmission power allocation. In addition, a dynamic resource allocation and task offloading (DRATO) algorithm is proposed [20], and we evaluate the performance of the DRATO algorithm through simulation experiments. Our main contributions are as follows:

(i) We consider a joint resource allocation and computation offloading for the multi-BS and multi-device-based NOMA system. The BSs are allocated to the MEC servers to provide computing resources for the IoT devices. When IoT devices transmit tasks to the MEC servers, we regard not only the interference of devices in the same BS using the same channel but also the interference of other BSs using the same channel.

(ii) We formulate the problem to minimize the energy consumption of all devices by jointly optimizing the computation resource allocation and transmission power allocation. We use the Lyapunov technology to transform problem and build model game to solve it. Firstly, based on Lyapunov theory, we divide our optimization problem into two subproblems. Then, one is solved by derivation, and the other is modeled as a noncooperative game model to solve.

(iii) We propose a DRATO algorithm to obtain the desired strategy sets of CPU-cycle frequency allocation and transmission power allocation. In addition, the effectiveness of our algorithm is verified by the parameter analysis and comparison experiments.

The rest of this paper is organized as follows. We build the system model and formulate the research problem in Section 2. The optimization problem is divided into two parts and we design DRATO algorithm in Section 3. In Section 4, the performance of DRATO algorithm is verified. The related work is presented in Section 5. Furthermore, we summarize this paper in Section 6.

2. System Model and Problem Formulation

In this section, we present the system model of resource allocation and computation offloading among the multi-BS and multidevice-based NOMA, while we also formulate the problem to minimize the energy consumption of all devices by jointly optimizing the computation resource allocation and transmission power allocation.

2.1. System Framework. As depicted in Figure 1, we consider a joint resource allocation and computation offloading system among multi-BS and multi-IoT-device-based NOMA. The BSs are equipped with MEC servers and denoted by a set $\mathcal{M} = \{1, 2, \ldots, M\}$. The IoT devices are defined by a set $\mathcal{I} = \{1, 2, \ldots, I\}$, which can process their tasks locally or offload them to MEC servers to process via several wireless channels. There is interference among devices sharing the same wireless channel. Based on [21], a discrete time-slotted model is considered; we define each time slot length as τ and $t \in \{0, 1, \ldots, T-1\}$. In addition, the main notations are listed in Table 1.

2.2. Local Processing Model. The tasks of IoT devices can be processed locally or offloaded to MEC servers to process. In the t-th time slot, the number of data bits computed locally for device i of BS m is given as follows:

$$B_{i,m}^{l}(t) = \tau \frac{f_{i,m}(t)}{c_i}, \quad \forall i \in I, m \in M, \tag{1}$$

where $f_{i,m}(t)$ represents the CPU-cycle frequency for device i of BS m. c_i denotes the required CPU-cycles for computing 1 bit of device i and τ is the time slot length.

In the t-th time slot, the computation energy consumption for local processing of device i of BS m is obtained as follows:

$$E_{i,m}^{l}(t) = k_i \left(f_{i,m}(t)\right)^2 c_i B_{i,m}^{l}, \tag{2}$$

where k_i is the effective switched capacitance, which depends on the chip structure of the IoT device i [22].

2.3. Task Offloading Model. We denote the transmission power and the transmission power constraint for IoT device i of BS m on channel n in t-th time slot as $p_{i,m,n}(t)$ and $p_{\max}$, respectively [23]. Thus, we have

$$0 \le p_{i,m,n}(t) \le p_{\max}. \tag{3}$$

In NOMA, a subchannel can be allocated to multiple devices in the same time slot t. Based on [19], the BS adopts SIC technique to eliminate the interference from the received signal. The devices with higher channel gains are decoded in turn and the other devices are seen as the interference. In time slot t, $g_{i,m,n}(t)$ represents the uplink channel gain from device i to BS m on channel n. We assume that the channel gain of each device is in an increasing order; that is, $g_{1,m,n}(t) \le g_{2,m,n} \le \ldots \le g_{i,m,n}(t) \le \ldots \le g_{I,m,n}$. The signal-to-interference-plus-noise ratio (SINR) of the signal received by device i is given as follows:

$$r_{i,m,n}(t) = \sum_{i\epsilon\left\{S_{m,n} | g_{u,m,n} < g_{i,m,n}\right\}} p_{u,m,n}(t) g_{u,m,n}(t), \tag{4}$$

where $S_{m,n}$ is the set of devices transmitting tasks to BS m on channel n. $i\epsilon\left\{S_{m,n} | g_{u,m,n} < g_{i,m,n}\right\}$ means that device i is in $S_{m,n}$ and its channel gain is greater than device u. $p_{u,m,n}(t)$ represents the transmission power of the IoT device i from BS m on channel n in t-th time slot [24, 25].

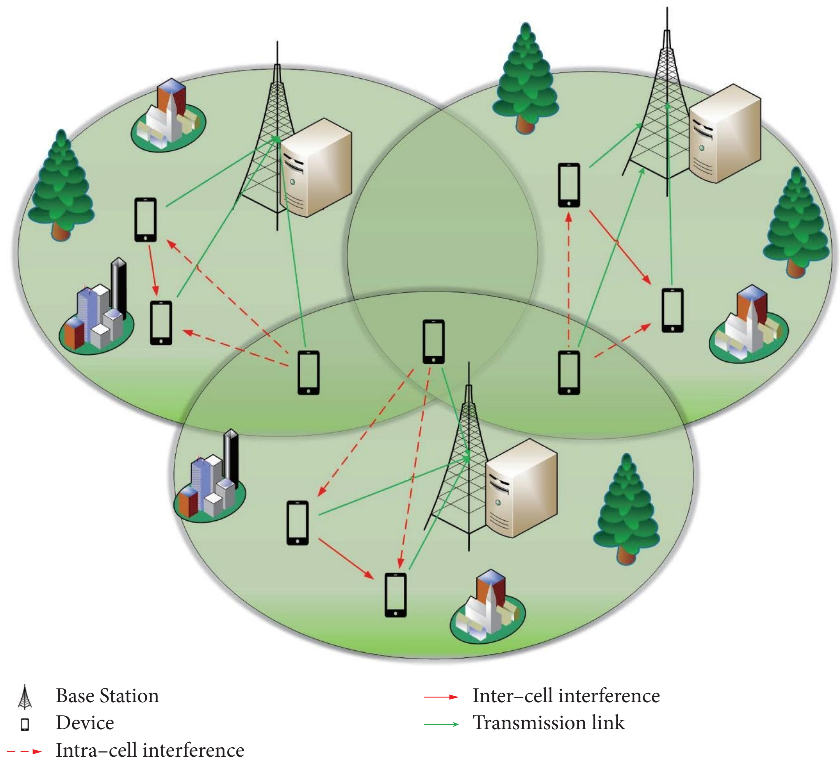

Figure 1: Multidevice system-based NOMA in MEC.

Table 1: Notations.

Symbol	Description
$\mathscr{I}$	The set of IoT devices
$\mathscr{M}$	The set of BSs
$\mathscr{N}$	The set of channels
τ	Time slot length
$f_{i,m}(t)$	The CPU-cycle frequency for device i of BS m
$f_{\max}$	The CPU-cycle frequency constraint for devices
c_i	The required CPU cycles to compute 1-bit data of device i
k_i	The effective switched capacitance of the CPU chip structure of device i
$B^l_{i,m}(t)$	The number of data bits computed locally for device i of BS m
$E^l_{i,m}(t)$	The energy consumed of local computation for device i of BS m
$p_{i,m,n}(t)$	The amount of transmit power of device i of BS m on channel n
$p_{\max}$	The transmission power constraint of devices
$R_{i,m}(t)$	The data transmission rate from device i to BS m
$B^r_{i,m}(t)$	The number of data bits transmitted from i-th device to BS m
$E^r_{i,m}(t)$	The energy consumption of data transmission from device i to BS m
$r_{i,m,n}(t)$	The interference for device i caused by other devices of same BS m on channel n
$\tilde{r}_{i,m,n}(t)$	The interference for device i caused by other devices of other BSs on channel n
$g_{i,m,n}(t)$	The channel gain for device i of BS m on channel n
Q_i	Queue length of i-th device
V	The trade-off parameter of queue length and energy consumption
B	The bandwidth of the devices
$E(t)$	Total energy consumption of devices
$S_{m,n}$	The set of devices transmitting tasks to BS m on channel n

On channel n, device i is not only interfered by devices in the same BS but also interfered by devices in other BSs on channel n. Thus, the received SINR of device i from BS v on channel n in time slot t is given as follows:

$$\widetilde{r}_{i,m,n}(t) = \sum_{v=1, v\neq m}^{M} \sum_{i=1}^{I} p_{i,v,n}(t) g_{i,v,n}(t). \tag{5}$$

Referring to Shannonformula, the data transmission rate for device i of BS m on the n-th channel at time slot t is shown as follows:

$$R_{i,m,n}(t) = B\log_2\left(1 + \frac{p_{i,m,n}(t) g_{i,m,n}(t)}{\sigma^2 + r_{i,m,n}(t) + \widetilde{r}_{i,m,n}(t)}\right), \tag{6}$$

where B is bandwidth of BSs. For each IoT device i, we define σ^2 as the noise power.

The energy consumed by offloading tasks from device i to BS m during time period τ is obtained as follows:

$$E_{i,m}^{r}(t) = \tau p_{i,m,n}(t). \tag{7}$$

If we denote $A_i(t)$ as the number of new generated tasks by the i-th device in time slot t and the tasks can be processed in the next time slot $t+1$, the queue length of the i-th IoT device's task buffer in time slot t is denoted by $Q_i(t)$. Thus, the queue length in time slot $t+1$ is as follows:

$$Q_i(t+1) = \max\left[Q_i(t) - B_{i,m}^{l}(t) - B_{i,m}^{r}(t), 0\right] + A_i(t), \tag{8}$$

where $B_{i,m}^{l}(t)$ represents the number of data bits computed locally for device i of BS m in t-th time slot. $B_{i,m}^{r}(t)$ is the number of data bits transmitted from device i to BS m in t-th time slot.

2.4. Problem Formulation. According to above local processing model and task offloading model, the total energy consumption is represented as

$$E(t) = \sum_{i=1}^{I} \left[E_{i,m}^{l}(t)\right] + E_{i,m}^{r}(t). \tag{9}$$

In this paper, the objective is to minimize the average total energy consumption of all devices by jointly optimizing computation resource allocation and transmission power allocation under CPU-cycle frequency constraint, transmission power constraint, and task buffers constraint. Then, the optimization problem can be formulated as follows:

$$\begin{aligned} &P1 \min_{f_{i,m}(t), p_{i,m,n}(t)} \overline{E}(t) = \lim_{T\longrightarrow\infty} \frac{1}{T}\sum_{t=1}^{T} E\left[\sum_{i=1}^{I}\left[E_{i,m}^{l}(t) + E_{i,m}^{r}(t)\right]\right] \text{s.t.},\\ &C_1 . 0 \leq f_{i,m}(t) \leq f_{\max},\\ &C_2 . 0 \leq p_{i,m,n}(t) \leq p_{\max},\\ &C_3 . \lim_{t\longrightarrow\infty} \frac{E[Q_i(t)]}{t} = 0. \end{aligned} \tag{10}$$

It is obvious that problem $P1$ is a stochastic optimization problem. It is because the new arrival tasks and the transmission power of IoT devices are difficult to predict, which are dynamic and stochastic. Thus, we take advantage of Lyapunov optimization technology, which can decompose the stochastic optimization problem into several deterministic optimization subproblems [26, 27].

3. Dynamic Resource Allocation and Task Offloading for IoT in MEC

3.1. Problem Transformation. According to Lyapunov optimization technology, the optimization problem (10) is transformed into two deterministic optimization subproblems [28]. We define $Z(t)$ as the queue backlog matrix, and $Z(t) = [Q_1(t), Q_2(t), \ldots, Q_n(t)]$. When $t=0$, $L(Z(t))=0$. The Lyapunov function is defined as

$$L(Z(t)) = \frac{1}{2}\sum_{i=1}^{I} Q_i^2(t). \tag{11}$$

Specifically, $L(Z(t))$ represents the queue backlog state of the IoT devices. If and only if the task queues' congestion is large, $L(Z(t))$ will be large. When the queue backlogs of IoT devices become small, $L(Z(t))$ will be small. Thus, we can reduce the value of $L(Z(t))$ to obtain the low congestion state of queue.

The *one*-slot conditional Lyapunov drift $\Delta(Z(t))$ is represented as follows:

$$\Delta(Z(t)) = E\{L(Z(t+1)) - L(Z(t)) | Z(t)\}. \tag{12}$$

Combining our optimization problem to minimize the transmission energy consumption and the IoT devices' queue length, we define the drift-plus-energy equation in each time slot as follows:

$$\Delta v(Z(t)) = \Delta(Z(t)) + VE[E|Z(t)], \tag{13}$$

where V is the trade-off parameter between the queue length and energy consumption. Without loss of generality, $V \geq 0$. The greater the value of V and the greater the weight of energy consumption, we give the upper bound of the drift-plus-energy.

Theorem 1. *For $0 \leq f_{i,m}(t) \leq f_{\max}$ and $0 \leq p_{i,m,n}(t) \leq p_{\max}$, the function $\Delta v(Z(t))$ satisfies the following inequality:*

$$\begin{aligned} \Delta(Z(t)) + VE[E(t)|Q(t)] &\leq \eta + VE[E(t)|Q(t)]\\ &+ E\left[\sum_{i=1}^{I} Q_i(t)\left[A_i(t) - B\sum_{i,m}(t)\right] Z(t)|\right], \end{aligned} \tag{14}$$

where η is a fixed value.

Proof. According to equation (8) and $\max[a-b, 0]^2 \leq (a-b)^2$ for $a, b > 0$, we let $\max[Q_i(t) - B_{i,m}^{l}(t) - B_{i,m}^{r}(t), 0] = Q_i(t) - B\sum_{i,m}(t)$, where $B\sum_{i,m}(t) = B_{i,m}^{l}(t) + B_{i,m}^{r}(t)$ or $B\sum_{i,m}(t) = Q_i(t)$. Thus, we have

$$Q_i(t+1)^2 = \left(\max\left[Q_i(t) - B^l_{i,m}(t) - B^r_{i,m}(t), 0\right]\right)^2 + A_i(t)^2 + 2A_i(t)\max\left[Q_i(t) - B^l_{i,m}(t) - B^r_{i,m}(t), 0\right] \leq \left(Q_i(t) - B\sum_{i,m}(t)\right)^2 + A_i(t)^2 + 2A_i(t)Q_i(t). \tag{15}$$

Subtracting $Q_i(t)^2$ from both sides of the inequality in (15) and dividing by 2, we have

$$\frac{1}{2}\left[Q_i(t+1)^2 - Q_i(t)^2\right] \leq \frac{1}{2}\left[B\sum_{i,m}(t)^2 + A_i(t)^2\right] + Q_i(t)\left[A_i(t) - B\sum_{i,m}(t)\right]. \tag{16}$$

Summarizing the above equations (11) and (12) and the inequality in (16), we obtain

$$\Delta(Z(t)) \leq \frac{1}{2}\sum_{i=1}^{I}\left[B\sum_{i,m}(t)^2 + A_i(t)^2\right] + \sum_{i=1}^{I} Q_i(t) E\left\{A_i(t) - B\sum_{i,m}(t) | Z(t)\right\}. \tag{17}$$

Finally, we add $VE(t)$ to both sides of equation (17) and get conditional expectation value, which is as follows:

$$\Delta(Z(t)) + VE\{E(t)|Z(t)\} \leq \frac{1}{2}E\left\{\sum_{i=1}^{I}\left[A_i(t)^2 + B\sum_{i,m}(t)^2 | Z(t)\right]\right\} + VE\{E(t)|Z(t)\} + E\left\{\sum_{i=1}^{I} Q_i(t)\left[A_i(t) - B\sum_{i,m}(t)\right] | Z(t)\right\}. \tag{18}$$

Note that $A_i(t) + B\sum_{i,m}(t)^2$ with condition $Z(t)$ is deterministic; hence,

$$E\left\{\sum_{i=1}^{I}\left[A_i(t)^2 + B\sum_{i,m}(t)^2 | Z(t)\right]\right\} = \sum_{i=1}^{I}\left[A_i(t)^2 + B\sum_{i,m}(t)^2\right]. \tag{19}$$

Defining $\eta = \sum_{i=1}^{I}[A_i(t)^2 + B\sum_{i,m}(t)^2]$, the proof of Theorem 1 is completed. □

3.2. Dynamic Resource Allocation and Task Offloading Algorithm Design. Because η and $A_i(t)$ are the constants in each slot, the optimization problem is rewritten as follows:

$$P2 \min_{f_{i,m}(t), p_{i,m,n}(t)} VE(t) - \sum_{i=1}^{I} Q_i(t) B \sum_{i,m}(t). \tag{20}$$

s.t.

$$\mathbf{C}_1 \cdot 0 \leq f_{i,m}(t) \leq f_{\max}, \tag{21}$$

$$\mathbf{C}_2 \cdot 0 \leq p_{i,m,n}(t) \leq p_{\max}. \tag{22}$$

Obviously, problem $P2$ can be divided into two parts to optimize. The optimization of the first part is related to local computing, and the main parameter of optimization is the device's CPU-cycle frequency strategy. The optimization of the second part is related to the task offloading. The optimization parameter is the transmission power strategy of the device offloading tasks to the MEC servers. Thus, problem (20) can be converted into two independent subproblems $P2.1$ and $P2.2$, and we will solve them separately.

3.2.1. Computation Resource Allocation. This part mainly studies the CPU-cycle frequency strategy optimization in local calculation.

$$P2.1 \min_{f_{i,m}(t)} \sum_{i=1}^{I} \tau\left[Vk_i f_{i,m}(t)^3 - Q_i(t)\frac{f_{i,m}(t)}{c_i}\right], \tag{23}$$

s.t.

$$0 \leq f_{i,m}(t) \leq f_{\max}. \tag{24}$$

Problem $P2.1$ is convex and the optimal solution is straightforward. The formula took the derivative with respect to $f_{i,m}(t)$, which is expressed as $3Vk_i(f_{i,m}(t))^2\tau - \tau Q_i(t)/c_i$. Firstly, we make the derivative be zero and get $f_{i,m}(t) = \sqrt{Q_i(t)/3Vk_ic_i}$. Then, we get its second derivative function as $6Vk_i f_{i,m}(t)\tau \geq 0$. Thus, we obtain the optimal solution of $f_{i,m}(t)$, which is as follows:

$$f_{i,m}(t) = \min\left\{\sqrt{\frac{Q_i(t)}{3Vk_ic_i}}, f_{\max}\right\}. \tag{25}$$

3.2.2. Transmission Power Allocation. This part mainly studies the optimization of transmission power strategy in task offloading process.

$$P2.2 \min_{p_{i,m,n}(t)} \sum_{i=1}^{I}\left[VE^r_{i,m}(t) - Q_i(t)B^r_{i,m}(t)\right], \tag{26}$$

s.t.

$$0 \leq p_{i,m,n}(t) \leq p_{\max}. \tag{27}$$

In the transmission power decision-making problem among multiple devices, the power decision of each device depends not only on its own power strategy but also on the power strategies of other devices. All devices want to choose the optimal power strategies to reduce owner energy consumption, so there is a competitive relationship of transmission power resources among devices. Game theory is a good tool to solve the problem of competition decision-making among players, which uses a mathematical model to solve the interest conflict among multiple players and obtain the best strategy set [29]. Thus, we can use game theory to address the problem of transmission power decision-making.

In this part, the transmission power decision-making problem among devices is modeled as a transmission power game. In this game, the players are all devices competing for power resources to minimize their owner energy consumption. The transmission power strategy game can be defined as $\Gamma = (\mathcal{I}, \{\mathrm{p}\}_{i\in\mathcal{I}}, \{F\}_{i\in\mathcal{I}})$, where $\mathcal{I}$ is the set of devices and p is the feasible strategy set of devices. The energy consumption function is F. Each device seeks a profitable power strategy to minimize its own energy consumption.

Formula (26) is simplified and expressed as

$$F\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right) = \tau\sum_{i=1}^{I}\left(V p_{i,m,n}(t) - BQ_i(t)\log_2 \cdot\left(1 + \frac{p_{i,m,n}(t)g_{i,m,n}(t)}{\sigma^2 + r_{i,m,n}(t) + \tilde{r}_{i,m,n}(t)}\right)\right). \tag{28}$$

Definition 1 (Nash Equilibrium). For a noncooperative game Γ, if there is a strategy set $p^* = (p_1^*, p_2^*, \ldots, p_I^*)$ and no device wants to reduce its own energy consumption by changing owner strategy, then we call p^* an NE strategy profile.

$$F_i\left(p_{i,m,n}(t)^*, p_{-i,m,n}(t)^*\right) \le F_i\left(p_{i,m,n}(t), p_{-i,m,n}(t)^*\right), \tag{29}$$

where F_i is device i's energy consumption and $p^*_{-i,m,n}$ presents the strategies chosen by devices excluding device i.

The NE is a stable state, in which no device has any intention to change its strategy for reducing energy consumption. Thus, we can obtain the optimal strategy set by proving the existence of NE. The definition of the exact potential game is given as follows.

Definition 2. (*Potential Game*) For Γ, if and only if there is a potential function $\Phi(p_{i,m,n}(t), p_{-i,m,n}(t))$, the relationship between $\Phi(p_{i,m,n}(t), p_{-i,m,n}(t))$ and $F_i(p_{i,m,n}(t), p_{-i,m,n}(t))$ is as follows:

$$\begin{aligned} &F_i\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right) - F_i\left(p'_{i,m,n}(t), p_{-i,m,n}(t)\right) = \\ &\Phi\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right) - \Phi\left(p'_{i,m,n}(t), p_{-i,m,n}(t)\right). \end{aligned} \tag{30}$$

Γ is an exact potential game, and there is at least a pure strategy NE, which should be proved by us. The specific certification process is as follows.

Theorem 2. *Γ is an exact potential game, and the potential function is as follows:*

$$\begin{aligned} \Phi\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right) &= \tau\left(V p_{i,m,n}(t)\right) \\ &-BQ_i(t)\log_2\left(p_{i,m,n}(t)g_{i,m,n}(t) + \sigma^2 + r_{i,m,n}(t) + \tilde{r}_{i,m,n}(t)\right). \end{aligned} \tag{31}$$

Proof. Based on formula (31), when devices' power strategies change, we get the following equation:

$$\begin{aligned} &F\left(p_{i,m,n}'(t), p_{-i,m,n}(t)\right) - F\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right) = \\ &V\tau\sum_{j\in\mathcal{I}/j\neq i} p_{j,m,n}(t) + V\tau p_{i,m,n}'(t) - \tau \\ &\sum_{j\in\mathcal{I}/j\neq i} Q_j(t)B\log_2\left(p_{j,m,n}(t)g_{j,m,n}(t) + \delta_j\right) \\ &-\tau Q_i(t)B\log_2\left(p_{i,m,n}'(t)g_{i,m,n}(t) + \delta_i\right) + \\ &\tau\sum_{j\in\mathcal{I}/j\neq i} Q_j(t)B\log_2\left(\delta_j\right) + \tau Q_i(t)B\log_2\left(\delta_i\right) - \\ &V\tau\sum_{j\in\mathcal{I}/j\neq i} p_{j,m,n}(t) - V\tau p_{i,m,n}(t) \\ &+\tau\sum_{j\in\mathcal{I}/j\neq i} Q_j(t)B\log_2\left(p_{j,m,n}(t)g_{j,m,n}(t) + \delta_j\right) \\ &+\tau Q_i(t)B\log_2\left(p_{i,m,n}(t)g_{i,m,n}(t) + \delta_i\right) \\ &-\tau\sum_{j\in\mathcal{I}/j\neq i} Q_j(t)B\log_2\left(\delta_j\right) - \tau Q_i(t)B\log_2\left(\delta_i\right) \\ &= V\tau p'_{i,m,n}(t) - V\tau p_{i,m,n}(t) \\ &\quad-\tau Q_i(t)B\log_2\left(p'_{i,m,n}(t)g_{i,m,n}(t) + \delta_i\right) \\ &\quad+\tau Q_i(t)B\log_2\left(p_{i,m,n}(t)g_{i,m,n}(t) + \delta_i\right), \end{aligned} \tag{32}$$

where $\delta_i = \sigma^2 + r_{i,m,n}(t) + \tilde{r}_{i,m,n}(t)$.

With the devices' power strategies changing, we get the change of the potential function as follows:

$$\begin{aligned} &\Phi\left(p'_{i,m,n}(t), p_{-i,m,n}(t)\right) - \Phi\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right) \\ &\quad= V\tau p'_{i,m,n}(t) - V\tau p_{i,m,n}(t) \\ &\quad-\tau Q_i(t)B\log_2\left(p_{i,m,n}'(t)g_{i,m,n}(t) + \delta_i\right) \\ &\quad+\tau Q_i(t)B\log_2\left(p_{i,m,n}(t)g_{i,m,n}(t) + \delta_i\right) \\ &= F\left(p'_{i,m,n}(t), p_{-i,m,n}(t)\right) - F\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right). \end{aligned} \tag{33}$$

Thus,

$$\begin{aligned} &F_i\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right) - F_i\left(p_{i,m,n}'(t), p_{-i,m,n}(t)\right) = \\ &\Phi\left(p_{i,m,n}(t), p_{-i,m,n}(t)\right) - \Phi\left(p_{i,m,n}'(t), p_{-i,m,n}(t)\right). \end{aligned} \tag{34}$$

Theorem 2 is proved. □

Summarizing the above two parts, we can determine the optimal computing resource allocation decisions and transmission power allocation decisions. The specific steps are described in Algorithm 1.

To effectively solve the optimization problem, we propose the dynamic resource allocation and task offloading (DRATO) algorithm. In the DRATO algorithm, we adopt the idea of best response to attain the optimal strategy set. In

Require:
(1) Input: $\mathcal{I}, \mathcal{M}, \mathcal{N}, \mathcal{S}_{m,n}, c_i, k_i, g_{i,m,n}(t), B, \sigma, A_i(t)$;
(2) Output: The CPU-cycle frequency decision $f^*(t)$ and transmission power decision $p^*(t)$;
(3) **for** $w \leftarrow 1$ to W **do**
(4) **for** $i \leftarrow 1$ to I **do**
(5) Compute $f_{i,m}(t)$ according to the formula (25);
(6) Compute $B^l_{i,m}(t)$ according to (1);
(7) $F_i(t)$ are calculated according to the formula (28);
(8) Obtain the power strategy $p_{i,m,n}(t)$ of device i with the minimal energy consumption $p_i(t) = \operatorname{argmin}_{p_{i,m,n} \in \mathrm{p}} F(p_{i,m,n}(t), p_{-i,m,n}(t))$;
(9) Compute $B^r_{i,m}(t) = \tau R_{i,m,n}(t)$ based on the known variable values;
(10) Update queue length $Q_i(t)$ according to (8);
(11) **end for**
(12) Each device i attains the best power strategies $p_{i,m,n}(t)$ in the w-th iterations;
(13) Generate the updated strategy profile, $p(w+1) = \{p_1(w+1), p_2(w+1), \ldots, p_I(w+1)\}$;
(14) **end for**

Algorithm 1: The dynamic resource allocation and task offloading (DRATO) algorithm.

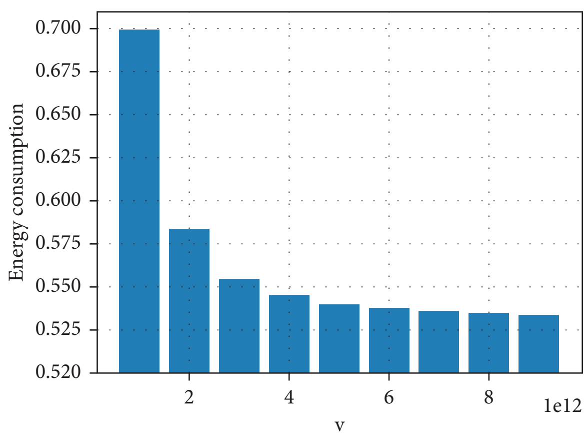

Figure 2: Effect of V on energy consumption.

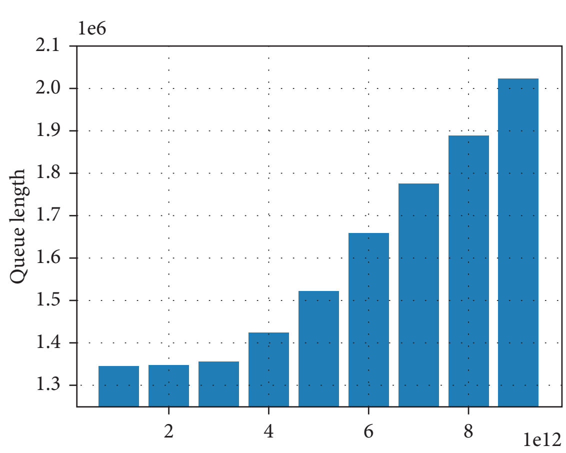

Figure 3: Effect of V on queue length.

steps 5–7, we obtain the values of $f_{i,m}(t)$, $B^l_{i,m}(t)$, and $F_i(t)$ based on initialized parameters. In step 8, we always select the transmission power strategy of device i with the minimal energy consumption in each iteration w. According to the power strategy obtained, we calculate $B^r_{i,m}(t)$ and update the queue length. After each iteration for all devices, we will update the policy set of the device and take it as the initial policy set for the next iteration. Because of the existence of NE, we can obtain the optimal strategy set through finite iterations. When the number of iteration is W, the iteration ends. Meanwhile, IoT devices get the lowest energy consumption and the optimal strategy set.

4. Simulation Results

This section consists of parameter analysis and comparison experiments. We have analyzed the influence of four parameters on the energy consumption and queue length. The performance of DRATO algorithm is evaluated by comparison experiments.

We consider a NOMA-MEC-enabled network with 3 BSs and 9 IoT devices. The IoT devices are randomly distributed to the BSs' coverage area. There are overlapping areas among BSs. The system bandwidth is available to all BSs, which is $1e6$ Hz [26]. The time slot of simulation results is 3000 s. Moreover, the simulation parameters used throughout the simulations are set in Table 2.

4.1. Parameter Analysis

(1) Impact of trade-off parameter V: Figures 2 and 3 describe the impact of trade-off parameter V on energy consumption and queue length. As shown in Figure 2, as V increases, the energy consumption will decrease. It is because the larger V is, the higher the weight of energy consumption is compared to the queue length. The DRATO algorithm reduces the energy consumption of the system by adjusting the allocation strategies of CPU-cycle frequency and transmission power. According to Figure 3, the queue length also increases with the rise of V. With the increase of V, the system always waits for the best time to process tasks. For example, devices backlog a

Table 2: Parameters used in the evaluation.

Parameters	Value
The number of devices	9
The number of BSs	3
The number of channels	9
The required CPU-cycles to compute 1-Bit data of devices	737.5 cycles/bit [23]
The max CPU-cycle frequency for devices	1e9 Hz [23]
The bandwidth of the BS	1e6 Hz
The transmission power for each device	[0.02, 0.1] W [22]
The background noise	$3.98e-21$ W [30]

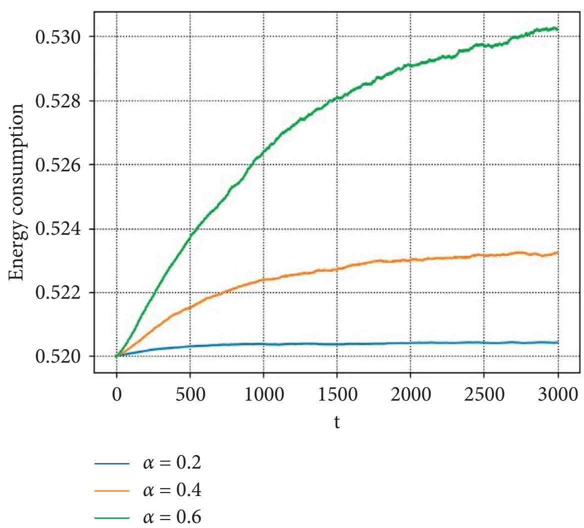

Figure 4: Effect of the task arrival rate on energy consumption.

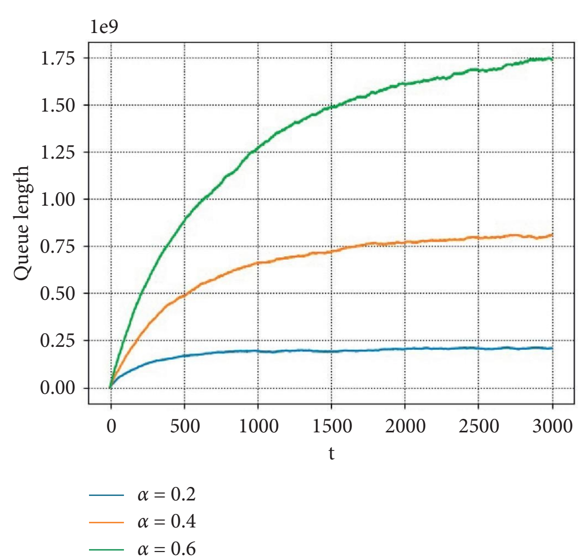

Figure 5: Effect of task arrival rate on queue length.

number of tasks locally and then offload the backlog of tasks to the edge servers to process in a specific time slot. It will lead to the queue backlog. Combined with Figures 2 and 3, it means that DRATO algorithm can achieve a trade-off between energy consumption and queue length by adjusting parameter V, which also can keep the stability of the queue.

(2) Impact of task arrival rate: Figures 4 and 5 describe the effect of task arrival rate on energy consumption and queue length. We consider that the task arrival rate is $\alpha \cdot A_i(t)$, and α is 0.2, 0.4, and 0.6, respectively. As shown in Figure 4, with the growth of task arrival rates, energy consumption has an upward trend. It is because, with more tasks, more tasks need to be computed locally or offloaded to MEC servers to process. Meanwhile, total energy consumption becomes large. According to Figure 5, when the task arrival rate increases, queue length will tend to rise. The reason is that the number of tasks increases and tasks are not handled in time, which will lead to a backlog of tasks and result in the increase of queue length.

(3) Impact of computing resource constraint $f_{\max}$: Figure 6 depicts the effect of computing resource constraint $f_{\max}$ on total energy consumption including energy consumption of local computation and energy consumption of task offloading process. As shown in Figure 6, when the CPU-cycle frequency rises, the energy consumption will increase accordingly. It is because local computing energy consumption increases with the increase of CPU-cycle frequency allocated for devices, which leads to higher total energy consumption for devices.

(4) Impact of transmission power constraint $p_{\max}$: Figure 7 shows the effect of transmission power constraint $p_{\max}$ on total energy consumption. With the increase of the transmission power constraint, the total energy consumption also shows an upward trend. The reason is that the larger the transmission power allocated, the larger transmission energy consumption during task offloading process. Meanwhile, the total energy consumption will increase.

4.2. Comparison Experiment. To further verify the effectiveness of the proposed DRATO algorithm, we compare the energy consumption and queue length with three benchmark algorithms which are stated as follows:

(i) *AllLocal*: In each time slot, all devices choose to compute all tasks locally.

(ii) *AllOffload*: In each time slot, all computation tasks of each device are offloaded to MEC servers to process.

(iii) *Random*: In each time slot, all devices randomly assign tasks to compute locally or offload to MEC servers for execution.

Figure 8 describes the energy consumption of four different algorithms in the fixed time slot. With the change of time, the energy consumption of AllOffload algorithm increases sharply, and energy consumptions of the other

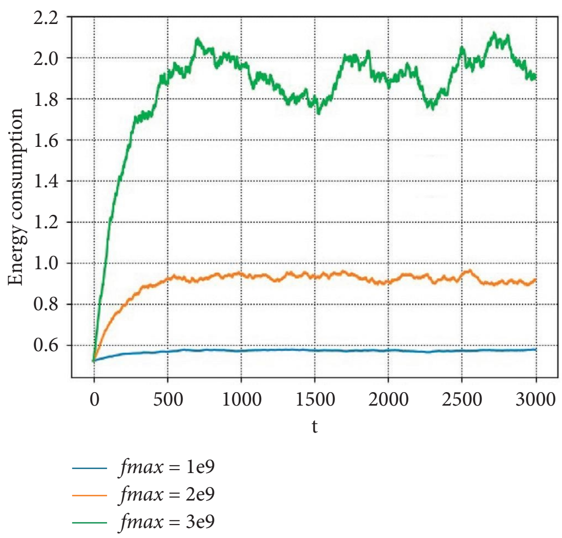

FIGURE 6: Effect of the computing resource constraint on energy consumption.

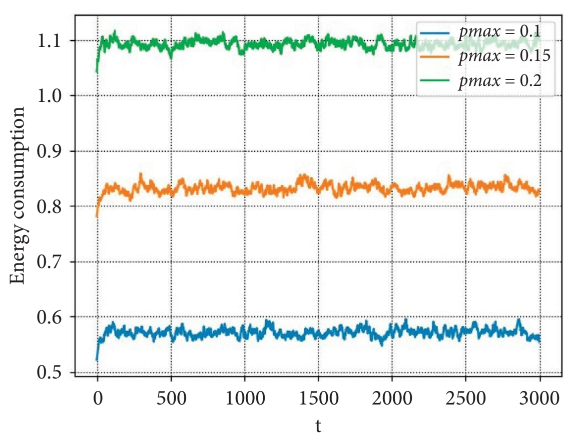

FIGURE 7: Effect of transmission power constraint on energy consumption.

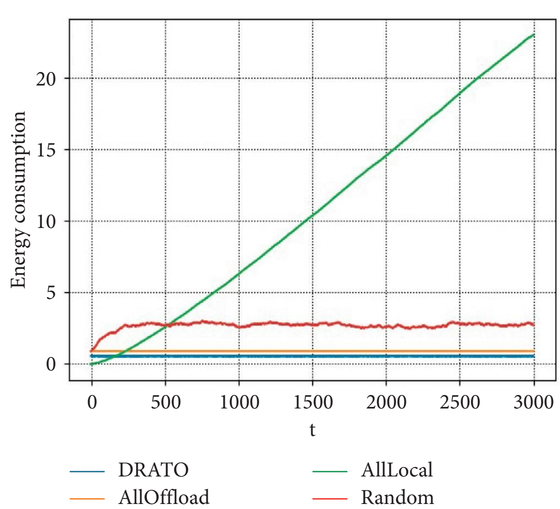

FIGURE 8: Performance comparison in terms of energy consumption.

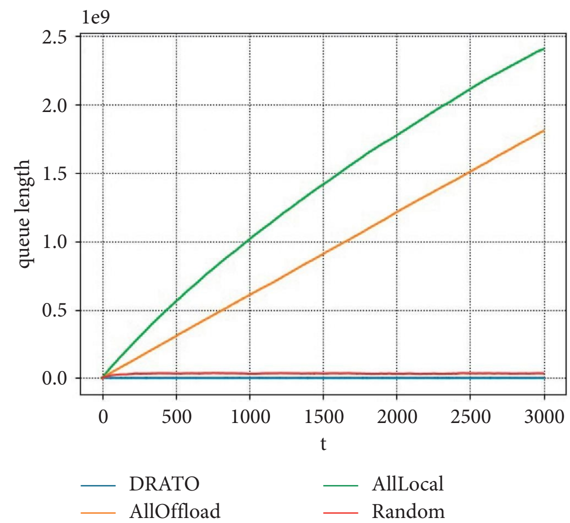

FIGURE 9: Performance comparison in terms of queue length.

three algorithms remain stabilized at a low value all the time. It can be observed that the energy consumption of the DRATO algorithm is lowest among four algorithms, which is 47% lower than that of AllOffload algorithm. It can be seen that the DRATO algorithm can effectively reduce the total energy consumption.

Figure 9 depicts the queue length of four different algorithms in the fixed time slot. Obviously, the queue length of the DRATO algorithm is lowest among four algorithms and it remains stable all the time. It is because the proposed DRATO algorithm can dynamically schedule the CPU-cycle frequency strategy and transmission power strategy to adapt to the changes of tasks arriving. As shown in Figure 9, AllOffload algorithm and AllLocal algorithm increase linearly with time. The reason is that the computation resources and transmission power resources of devices are limited. When the number of tasks is too large, there will be a backlog of tasks and the queue length will continue to increase. In the meantime, we can see that the queue lengths of the DRATO algorithm and the random algorithm remain at a low level.

Together with Figures 8 and 9, the proposed DRATO algorithm can effectively reduce the energy consumption and queue length, which has better performance.

5. Related Work

A number of research works focused on resource allocation and computing offloading in MEC in recent years [22, 31–35]. In [31], the authors jointly optimized the offline mode, channel allocation, and device-to-device (D2D) pairing to minimize the computational pressure of large-scale computing in MEC. They modeled the problem as the computing offloading task game among multiple users. Yu et al. considered the edge computing system-based NOMA in the power Internet of Things (PIoT). They exploited the Lyapunov technology to decompose the optimization problem into three subproblems: task splitting, resource block allocation, and computation resource allocation [32]. Zhao et al. studied a dynamic optimization problem of minimizing energy consumption and computing resources in MEC and decomposed the problem into four subproblems to solve based on the Lyapunov technology [22]. In [33], the authors considered joint edge computing and the cloud computing to achieve secure task offloading based blockchain scenario. They aimed to minimize energy consumption and the response time of task.

The NOMA technology can effectively improve spectral efficiency, which has been widely studied. Liu et al. considered uplink resource allocation NOMA-based scenario in 6G. They focused on maximizing the average total transmission rate of 6G-enabled cognitive IoT (CIoT) under the minimum transmission rates constraint [36]. In [37], the authors studied the intelligent uplink resource allocation based-NOMA system. They adapted the deep reinforcement learning (DRL) and SARSA-learning to design an effective algorithm. Liu et al. focused on the radio resource allocation and computation allocation-based NOMA for IoT devices scenario. They maximized energy efficiency by adjusting the strategy of the subchannel allocation and power allocation [38].

For the stochastic optimization problem, a number of works use Lyapunov method and reinforcement learning method [39–41]. In [39], the authors formulated a nonlinear integral dynamic optimization problem to reduce the cost and number of fresh sensors needed. They firstly transformed the optimization problem to the static problem. Then, they conducted a game model to solve problems, which can effectively reduce the complexity of the algorithm. Based on deep learning, Gu et al. proposed a job resource demand estimation method based on regression model to avoid overallocation of computing resources.

Different from existing research work, we study the computation resource allocation and transmission power allocation in combining resource allocation and computing offloading scenario. We formulate the stochastic problem to minimize the total energy consumption of IoT devices and apply the Lyapunov technology to transform it into two subproblems. In addition, we introduce the NOMA technology to eliminate the interinterference and intra-interference of IoT devices.

6. Conclusion

In this paper, we investigate the computing resource allocation and transmission power allocation to minimize total energy consumption for multidevice-based NOMA in MEC. Based on Lyapunov theory, we transform optimization problem into two deterministic subproblems and build the noncooperative game model to solve. A dynamic resource allocation and task offloading (DRATO) algorithm has been proposed to obtain the desired strategy sets of CPU-cycle frequency and transmission power. In addition, the effectiveness of our algorithm is verified by the simulation results.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

All authors have no conflicts of interest.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China (No. 61902029), R&D Program of Beijing Municipal Education Commission (No. KM202011232015), Project for Acceleration of University Classification Development (Nos. 5112211036, 5112211037, 5112211038).

References

[1] Z. Ding, P. Fan, and H. V. Poor, "Impact of impact of non-orthogonal multiple access on the offloading of mobile edge computing on-orthogonal multiple access on the offloading of mobile edge computing," *IEEE Transactions on Communications*, vol. 67, no. 1, pp. 375–390, 2019.

[2] L. Qi, W. Lin, X. Zhang, W. Dou, X. Xu, and J. Chen, "A correlation graph based approach for personalized and compatible web APIs recommendation in mobile APP development," *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[3] J. Li, H. Peng, Y. Cao et al., "Higher-order attribute-enhancing heterogeneous graph neural networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 1, 2021.

[4] J. Huang, M. Wang, Y. Wu, Y. Chen, and X. Shen, "Distributed offloading in overlapping areas of mobile edge computing for internet of things," *IEEE Internet of Things Journal*, vol. 9, no. 15, pp. 13837–13847, 2022.

[5] Y. Chen, Z. Liu, Y. Zhang, Y. Wu, X. Chen, and L. Zhao, "Deep reinforcement learning-based dynamic resource management for mobile edge computing in industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 4925–4934, 2021.

[6] J. Huang, Z. Tong, and Z. Feng, "Geographical POI recommendation for Internet of Things: a federated learning approach using matrix factorization," *International Journal of Communication Systems*, 2022.

[7] D. Zhou, X. Xue, and Z. Zhou, "SLE2: the improved social learning evolution model of cloud manufacturing service ecosystem," *IEEE Transactions on Industrial Informatics*, vol. 1, 2022.

[8] S. Fu, J. Gao, and L. Zhao, "Collaborative multi-resource allocation in terrestrial-satellite network towards 6G," *IEEE Transactions on Wireless Communications*, vol. 20, no. 11, pp. 7057–7071, 2021.

[9] H. Wu, Z. Zhang, C. Guan, K. Wolter, and M. Xu, "Collaborate edge and cloud computing with distributed deep learning for smart city internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8099–8110, 2020.

[10] G. Qu, H. Wu, R. Li, and P. Jiao, "A deep meta reinforcement learning-based task offloading framework for edge-cloud computing," *IEEE Transactions on Network and Service Management*, vol. 18, no. 3, pp. 3448–3459, 2021.

[11] Y. Xie, Y. Sheng, M. Qiu, and F. Gui, "An adaptive decoding biased random key genetic algorithm for cloud workflow scheduling," *Engineering Applications of Artificial Intelligence*, vol. 112, Article ID 104879, 2022.

[12] Y. Chen, W. Gu, and K. Li, "Dynamic task offloading for internet of things in mobile edge computing via deep

reinforcement learning," *International Journal of Communication Systems*, 2022.
[13] M.-S. Li, N. Cheng, J. Gao, Y. Wang, L. Zhao, and X. Shen, "Energy-efficient UAV-assisted mobile edge computing: resource allocation and trajectory optimization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3424–3438, 2020.
[14] S. Fu, J. Gao, and L. Zhao, "Integrated resource management for terrestrial-satellite systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3256–3266, 2020.
[15] L. Qi, C. Hu, X. Zhang, and M. R. Khosravi, "Privacy-aware data fusion and prediction with spatial-temporal context for smart city industrial environment," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4159–4167, 2021.
[16] L. Qi, Y. Yang, X. Zhou, W. Rafique, and J. Ma, "Fast anomaly identification based on multi-aspect data streams for intelligent intrusion detection toward secure industry 4.0," *IEEE Transactions on Industrial Informatics*, vol. 23, 2021.
[17] X. Xue, S. Wang, L. Zhang, Z. Feng, and Y. Guo, "Social learning evolution (SLE): computational experiment-based modeling framework of social manufacturing," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3343–3355, 2019.
[18] N. Ye, X.-M. Li, H. Yu, L. Zhao, W. Liu, and X. H. Deep, "NOMA: a unified framework for NOMA using deep multitask learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2208–2225, 2020.
[19] Y. Chen, H. Xing, Z. Ma, X. Chen, and J. Huang, "Cost-Efficient Edge Caching for NOMA-Enabled IoT Services," China Communications, 2022.
[20] Y. Xie, F. X. Gui, W. J. Wang, and C. F. Chien, "A two-stage multi-population genetic algorithm with heuristics for workflow scheduling in heterogeneous distributed computing environments," *IEEE Transactions on Cloud Computing*, vol. 3, 2021.
[21] W. U. Khan, F. Jameel, T. Ristaniemi, S. Khan, G. A. S. Sidhu, and J. Liu, "Joint spectral and energy efficiency optimization for downlink NOMA networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 2, pp. 645–656, 2020.
[22] F. Zhao, Y. Chen, Y. Zhang, Z. Liu, and X. Chen, "Dynamic offloading and resource scheduling for mobile-edge computing with energy harvesting devices," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 2154–2165, 2021.
[23] N. Nouri, A. Entezari, J. Abouei, M. Jaseemuddin, and A. Anpalagan, "Dynamic power–latency tradeoff for mobile edge computation offloading in NOMA-based networks," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2763–2776, 2020.
[24] H. Dai, Y. Zhao, G. Chen et al., "Robustly safe charging for wireless power transfer," *IEEE Transactions on Mobile Computing*, vol. 21, no. 6, pp. 2180–2197, 2022.
[25] H. Dai, X. Wang, X. Lin et al., "Placing wireless chargers with limited mobility," *IEEE Transactions on Mobile Computing*, 2021.
[26] Y. Chen, F. Zhao, X. Chen, and Y. Wu, "Efficient multi-vehicle task offloading for mobile edge computing in 6G networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 5, pp. 4584–4595, 2022.
[27] J. Huang, C. Zhang, and J. Zhang, "A multi-queue approach of energy efficient task scheduling for sensor hubs," *Chinese Journal of Electronics*, vol. 29, pp. 242–247, 2020.
[28] J. Huang, B. Lv, Y. Wu, Y Chen, and X Shen, "Dynamic admission control and resource allocation for mobile edge computing enabled small cell network," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 2, pp. 1964–1973, 2022.
[29] Y. Ding, K. Li, C. Liu, and K. Li, "A potential game theoretic approach to computation offloading strategy optimization in end-edge-cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 6, pp. 1503–1519, 2022.
[30] H. Zhang, B. Wang, C. Jiang et al., "Energy efficient dynamic resource optimization in NOMA system," *IEEE Transactions on Wireless Communications*, vol. 17, no. 9, pp. 5671–5683, 2018.
[31] T. Fang, F. Yuan, L. Ao, and J. Chen, "Joint task offloading, D2D pairing, and resource allocation in device-enhanced MEC: a potential game approach," *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3226–3237, 2022.
[32] H. Yu, Z. Zhou, Z. Jia, X. Zhao, L. Zhang, and X. Wang, "Multi-timescale multi-dimension resource allocation for NOMA-edge computing-based power IoT with massive connectivity," *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 3, pp. 1101–1113, 2021.
[33] H. Wu, K. Wolter, P. Jiao, Y. Deng, Y. Zhao, and M. Xu, "EEDTO: an energy-efficient dynamic task offloading algorithm for blockchain-enabled IoT-edge-cloud orchestrated computing," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2163–2176, 2021.
[34] Y. Chen, F. Zhao, Y. Lu, and X. Chen, "Dynamic task offloading for mobile edge computing with hybrid energy supply," *Tsinghua Science and Technology*, 2021.
[35] J. Xu, D. Li, W. Gu, and Y. Chen, "UAV-Assisted Task Offloading for IoT in Smart Buildings and Environment via Deep Reinforcement learning," *Building and Environment*, vol. 26, Article ID 109218, 2022.
[36] X. Liu, H. Ding, and S. Hu, "Uplink resource allocation for NOMA-based hybrid spectrum access in 6G-enabled cognitive internet of things," *IEEE Internet of Things Journal*, vol. 8, no. 20, pp. 15049–15058, 2021.
[37] W. Ahsan, W. Yi, Z. Qin, Y. Liu, and A. Nallanathan, "Resource allocation in uplink NOMA-IoT networks: a reinforcement-learning approach," *IEEE Transactions on Wireless Communications*, vol. 20, no. 8, pp. 5083–5098, 2021.
[38] B. Liu, C. Liu, and M. Peng, "Resource allocation for energy-efficient MEC in NOMA-enabled massive IoT networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 4, pp. 1015–1027, 2021.
[39] X. Ma, A. Zhou, Q. Sun, and S. Wang, "Freshness-aware information update and computation offloading in mobile-edge computing," *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 13115–13125, 2021.
[40] R. Gu, Y. Chen, S. Liu et al., "Liquid: intelligent resource estimation and network-efficient scheduling for deep learning jobs on distributed GPU clusters," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, 2021.
[41] R. Gu, K. Zhang, Z. Xu et al., "Fluid: Dataset Abstraction and Elastic Acceleration for Cloud-Native Deep Learning Training Jobs," *The 38th IEEE International Conference on Data Engineering*, vol. 3, pp. 2183–2196, 2022.