Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 3391917, 15 pages
https://doi.org/10.1155/2022/3391917

Research Article

Game-Theoretical Data Offloading for NOMA-Enabled Urban Internet of Things

Jintao Hu,[1] Jiajie Xu,[1] Jie Zhao,[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 28 May 2022; Revised 9 July 2022; Accepted 12 September 2022; Published 8 October 2022

Academic Editor: Geng Sun

Urban Internet of Things (IoT) plays an extremely important role in our daily life by deploying smart cities and urban brains. Orthogonal multiple access (OMA) technology has been a commonly used communication method in recent years, but nonorthogonal multiple access (NOMA) attracts the attention of many researchers due to its superiority of successive interference cancellation (SIC) technology. We consider adding the base station (BS) and unmanned aerial vehicle (UAV) to perform collaborative data offloading services with urban IoT devices and introduce the NOMA technology to improve offloading efficiency. In order to solve the data unloading problem in this model cost-effectively, we formulate the model as a game model based on noncooperative competition and propose the iterative game-based data offloading algorithm (GDOA) to obtain the Nash equilibrium (NE) solution. Finally, we use the simulation data to conduct parametric analysis experiments and comparison experiments on GDOA to evaluate its real performance.

1. Introduction

With the advancement of mobile communication technology, the urban IoT will play an increasingly important role in our daily life [1]. Different from ordinary IoT, urban IoT focuses on intelligent communication and smart city [2–4]. At the same time, the massive data generated by various sensors in the urban IoT puts enormous pressure to the IoT services, mainly including high communication delay, nonnegligible energy consumption, and quality of experience for urban IoT devices [5, 6]. In order to improve the quality of urban IoT communication services, it is essential to establish a reasonable framework, such as an efficient data offloading model [7–9]. Among them, the devices in the urban IoT can request services with the neighboring BS through wireless transmission, so as to determine the way of data offloading, i.e., the devices upload its data to be calculated in the server through the wireless link [10, 11]. However, since the data transmission in most urban IoT systems adopts the OMA technology, each device will be subject to a lot of intracell interference, which causes the delay and energy consumption of the device to become extremely high [12, 13]. In addition, the UAV is often used to assist the BS to perform data offloading services with devices, which reduces the computational pressure of the BS to a certain extent [14].

Nevertheless, due to the increasing data transmission and service requests in the urban IoT, the OMA technology is gradually unable to meet the services with a very large number of devices [15, 16]. Therefore, the NOMA technology that effectively reduces the intracell interference in the multirequest low-latency problem has been widely used [17, 18]. The main idea of NOMA is to send the nonorthogonal data from the transmitter and eliminate the interference at the receiver, and then, the realization of the whole process depends on the SIC technology [19–21]. The process of SIC can be expressed as follows: the receiving end judges the signals of the device one by one according to the descending order of channel gain, subtracts the signals of other devices whose channel gain is higher than that of the device from the interference, and finally loops the above operation until the successive interference of all devices is

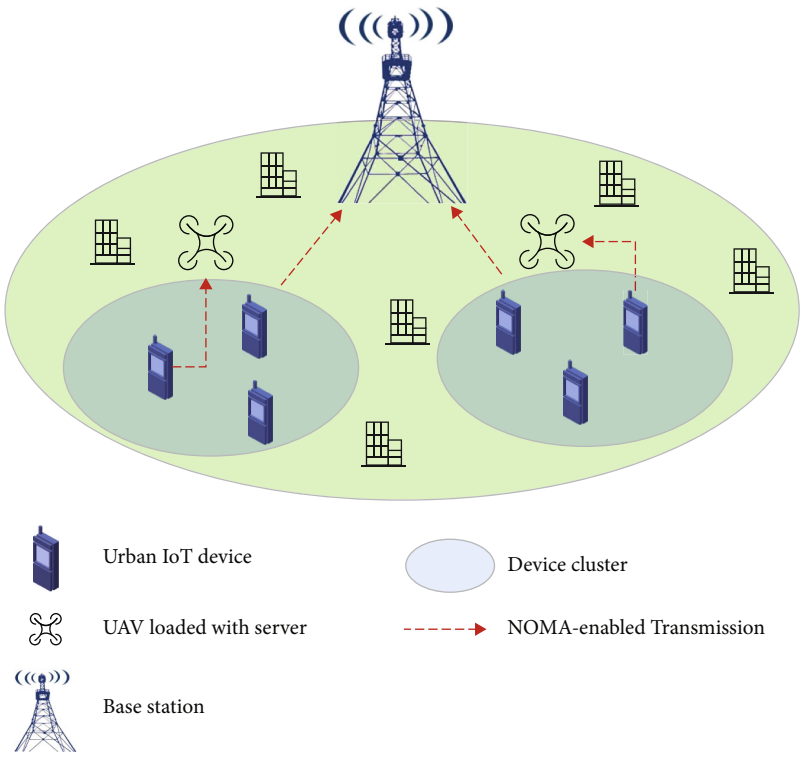

Figure 1: An example of NOMA-enabled data offloading scenario for urban IoT.

eliminated [22–24]. Because most of the interference is eliminated, NOMA greatly improves the efficiency of data offloading from devices in the urban IoT system.

In this paper, the scenario we consider is an urban IoT data offloading system that supports NOMA, the data offloading problem in this system is solved by the potential game method in game theory [25], and we add UAVs to assist the BS with the devices to communicate. Our optimization goal is to minimize the energy consumption cost of all urban IoT devices. According to the limited resource competition between devices, we formulate this model into a noncooperative competitive game model and propose a game-based data offloading algorithm (GDOA) based on the iterative update process, which can obtain the device strategy profile in NE state after a finite number of iterations [26]. Finally, a series of simulation performance evaluations are carried out for our proposed GDOA, including parameter analysis experiments and comparison experiments. The main contributions in our study include the following:

(i) We study a NOMA-enabled urban IoT data offloading model, while incorporating server-loaded UAVs to assist the BS in wireless communication with devices. There are three ways to offload data from devices: local computing, data offloading by the BS, or data offloading by the UAVs. Our overall optimization goal is to minimize the sum of the energy consumption costs of all urban IoT devices

(ii) We formulate this urban IoT data offloading model as a game model based on noncooperative competition between devices. The potential function in game theory is proposed, and it is proved that our game model is a generalized ordinal potential game. And we propose a game-based iterative algorithm GDOA, which can obtain the NE strategy profile after a finite number of iterations

(iii) We evaluate the simulation performance of this model, including many parameter analysis experiments of GDOA and several comparison experiments between GDOA and other excellent algorithms

The rest of this paper is organized as follows. The system model and problem formulation are introduced in Section 2. We describe the game-based data offloading for urban IoT in Section 3, including game-based offloading model and offloading algorithm design. And several parametric analysis experiments and comparison experiments are conducted in Section 4 to evaluate the performance of this model. Section 5 describes the related work. Finally, we summarize this study in Section 6.

2. System Model and Problem Formulation

2.1. System Model for Urban Internet of Things. As shown in Figure 1, we study a NOMA-based data offloading system for urban IoT, which consists of a single base station (BS), multiple UAVs, and multiple urban mobile devices. In this system, we consider n densely distributed urban mobile devices, and the set of all devices is $\mathcal{U} = \{u_1, u_2, \cdots, u_n\}$.

Table 1: Key notations.

Notations	Definitions
u_i	Any urban IoT device
$\mathcal{U}$	Set of all urban IoT devices
n	The number of all urban IoT devices
$\mathcal{W}$	Set of all urban IoT clusters
x	The number of urban IoT clusters
$\mathcal{V}$	Set of all UAVs
w_j	Set of all urban IoT devices in cluster j
y	The number of urban IoT devices in each cluster
v_j	The UAV in cluster j
f_i	The computing power of urban IoT device u_i
p_i	The transmit power of urban IoT device u_i
g_i	The channel gain of urban IoT device u_i for data transmission
B	The bandwidth of the channel for data transmission
K_i	The computation task of urban IoT device u_i
D_i	The data size of computation task K_i
X_i	The number of CPU cycles to complete computation task K_i
a_i	The data offloading strategy of urban IoT device u_i
$\mathbf{a}$	The set of all devices' data offloading strategies
a_{-i}	The set of all devices' data offloading strategies except device u_i
$\mathcal{S}_j$	The set of urban IoT device u_i's neighbor base station
$\mathcal{V}_j$	The set of all urban IoT devices allocated to channel c_j^k on edge server s_j
T_i^{loc}	The delay of completing computation task K_i locally
E_i^{loc}	The energy consumption of completing computation task K_i locally
T_i^{bs}	The delay of completing computation task K_i by the BS
E_i^{bs}	The energy consumption of completing computation task K_i by the BS
T_i^{uav}	The delay of completing computation task K_i by UAV
E_i^{uav}	The energy consumption of completing computation task K_i by UAV
$C_{a_{-i}}(a_i)$	The cost function of device u_i's strategy a_i
$\mathbf{a}^*$	The data offloading strategies of all devices at Nash equilibrium

There are x clusters covering y devices; the set of these clusters is expressed as $\mathcal{W} = \{w_1, w_2, \cdots, w_x\}$. And we represent each cluster as $w_j = \{u_1, u_2, \cdots, u_y\}$; thus, the set of all n devices can also be denoted by $\mathcal{U} = w_1 \cup w_2 \cdots \cup w_x$. There is a single base station S in our system; the devices of any cluster are within the coverage of this base station. The set of all UAVs is denoted as $\mathcal{V} = \{v_1, v_2, \cdots, v_x\}$, and one UAV v_j is arranged in each cluster w_j. The main notations and their definitions of this paper are given in Table 1.

The computing power of each device u_i when computing locally is f_i, which is related to device u_i's chip architecture. In addition, the transmit power of each device u_i is p_i, the bandwidth of the BS's channel and the UAV's channel is B, and the channel gain in data transmission for device u_i is g_i. We assume that device u_i will choose a data offloading approach (local computing, BS or UAV) based on its own resource requirements and possible cost.

To simplify the data transmission and computation of the device, each device u_i has a computation task $K_i \triangleq (D_i, X_i)$, where D_i represents the amount of intensive data when transmitting the task and X_i represents the number of CPU cycles required to compute the task [27].

Definition 1 (data offloading strategy). Each device u_i in this system needs to make a data offloading strategy $a_i = \{0, 1, 2\}$, $a_i = 0$ means that u_i chooses completing its computation task locally, $a_i = 1$ denotes that u_i chooses offloading the data of its computation task by BS, and $a_i = 2$ represents that u_i chooses offloading the data of its computation task by UAV.

Each cluster w_j shares one channel of the BS, and the set of devices sharing the same channel of the BS is denoted by

$$\mathcal{S}_j = \{u_i \in w_j \mid a_i = 1\}, \forall w_j \in \mathcal{W}. \quad (1)$$

And each cluster w_j shares one channel of the UAV; the set of devices sharing the same channel of the UAV is expressed as

$$\mathcal{V}_j = \{u_i \in w_j \mid a_i = 2\}, \forall w_j \in \mathcal{W}. \quad (2)$$

The strategy profile of all the devices is $\mathbf{a} = \{a_1, a_2, \cdots, a_n\}$, and $a_{-i} = \{a_1, \cdots, a_{i-1}, a_{i+1}, \cdots, a_n\}$ represents the strategy profile of all devices except device u_i.

2.2. NOMA-Enabled Transmission Model. In our system, the geographic positions of devices, BS, and UAVs are represented by three-dimensional coordinates, the position of device u_i is denoted by $L_i = (x_i, y_i, 0)$, the position of BS's server is represented by $H_m = (x_m, y_m, z_m)$, and UAV v_j server's position is expressed as $Q_j = (x_j, y_j, z_j)$. The distance between the device and the server can be calculated through their coordinates, and we can get the channel gain when device u_i chooses UAV v_j to offload its data, which is expressed as

$$g_{i,j} = \left\| L_i - Q_j \right\|^{-2}, \quad (3)$$

where $\|L_i - Q_j\| = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$.

We represent θ as the channel fading factor obeying the Rayleigh distribution, similar to the channel gain $g_{i,j}$; the channel gain when device u_i chooses BS S to offload its data is denoted by

$$g_{i,m} = \|L_i - H_m\|^{-\theta}, \quad (4)$$

where $\|L_i - H_m\| = \sqrt{(x_i - x_m)^2 + (y_i - y_m)^2 + z_m^2}$. Therefore, we can get the channel gain of device u_i in two cases, which can be expressed as

$$g_i = \begin{cases} g_{i,m}, & a_i = 1, \\ g_{i,j}, & a_i = 2. \end{cases} \quad (5)$$

Taking the BS server as an example, on the basis of the definition for NOMA at the transmitter, NOMA can broadcast the signal q_i of each device $u_i \in \mathcal{S}_j$ on the same channel, and the server of the BS receives a superposition-coded signal q_j^{bs} mixed with these signals, which can be expressed as

$$q_j^{bs} = \sum_{u_i \in \mathcal{S}_j} \sqrt{p_i} q_i. \quad (6)$$

Similarly, the superposition-coded signal q_j^{uav} received by the UAV server can be denoted by

$$q_j^{\text{uav}} = \sum_{u_i \in \mathcal{V}_j} \sqrt{p_i} q_i. \quad (7)$$

Then, we will introduce successive interference cancellation (SIC) technology in this NOMA-enabled data offloading system. SIC is a signal decoding method; its basic principle is to gradually subtract the interference of the device who possesses the maximum signal power. During this process, the SIC detector makes data judgments for multiple devices one by one in the received signal, and when a device is judged, the multiple access interference caused by the device's signal is subtracted at the same time, and the operation is performed in the order of signal power, and the signal with higher power is operated first. The operation is repeated until all the multiple access interference are eliminated [28]. In this order, any device u_i can correctly decode the superimposed signals of other devices whose channel gain is greater than g_i in the same channel, so as to eliminate successive interference in u_i's received signal.

Given the BS selection set $\mathcal{S}_j$ and UAV selection set $\mathcal{V}_j$, all devices on the same channel are sorted based on their channel gains, and the devices with higher channel gains are ranked ahead of the devices with lower channel gains. Thus, we can get the sorted sequence of all devices on the same channel of BS S: $u_1, u_2, \cdots, u_{|\mathcal{S}_j|}$, where $|\mathcal{S}_j|$ is the number of devices in $\mathcal{S}_j$, device u_1 and, the channel gain g_1 of device u_1 is the maximum value, the channel gain $g_{|\mathcal{S}_j|}$ of device $u_{|\mathcal{S}_j|}$ is the minimum value. And we can also obtain the sorted sequence of all devices on the same channel of UAV v_j: $u_1, u_2, \cdots, u_{|\mathcal{V}_j|}$, where $|\mathcal{V}_j|$ is the number of devices in $\mathcal{V}_j$, device u_1 and, the channel gain g_1 of device u_1 is the maximum value, the channel gain $g_{|\mathcal{V}_j|}$ of device $u_{|\mathcal{V}_j|}$ is the minimum value [29, 30].

We next investigate the SIC process between the device and the UAV server when the device selects the UAV to offload data. For two devices $u_i, u_b \in \mathcal{V}_j$, suppose u_b's channel condition is better than u_i, i.e., $g_b < g_i$. In order to use SIC reasonably, the data rate of device u_b for decoding device u_i's received signal cannot be lower than device u_i's data rate for decoding device u_i's received signal, and $r_{b|i}^{\text{uav}} \geq r_{i|i}^{\text{uav}}$. If the above condition cannot be satisfied, device u_i's data rate will reduce because its received inside interference is not eliminated. On the contrary, device u_i's data rate can be denoted by

$$r_i^{\text{uav}} = \min \left\{ r_{b|i}^{\text{uav}} \mid g_b < g_i \right\}, \forall u_b \in \mathcal{V}_j, \quad (8)$$

where $r_{b|i}^{\text{uav}}$ denotes device u_b's data rate for decoding device u_i's signal, which can be expressed as

$$r_{b|i}^{\text{uav}} = B\log_2 \left(1 + \frac{p_b g_b}{\sum_{t=i+1}^{|\mathcal{V}_j|} p_t g_t + \sigma^2} \right). \quad (9)$$

In other words, we can deduce that the data rate of device u_i is lower than the devices ranked after device u_i.

Through the above description, the decoding order of these devices derived from the channel gain is very important and directly affects the data rate of each device [31]. Therefore, how to get a reasonable order is essential when optimizing the data rate of all devices in this model. According to (8) and (9), we can obtain the available data rate r_i^{uav} of device u_i:

$$r_i^{\text{uav}} = B\log_2\left(1 + \frac{p_b}{\sum_{t=i+1}^{|\mathscr{V}_j|} p_t g_t + M_i}\right), \tag{10}$$

where M_i is denoted by

$$M_i = \max\left\{\frac{\sigma^2}{g_b} \mid g_b < g_i\right\}, \forall u_b \in \mathscr{V}_j. \tag{11}$$

In order to ensure that all devices can accept the data rate under the same transmit power, we should determine the decoding order according to the channel gain and the received intracell interference of the device which can be expressed as

$$M_1 = \frac{\sigma^2}{g_1} \geq M_2 = \frac{\sigma^2}{g_2} \geq \cdots \geq M_{|\mathscr{V}_j|} = \frac{\sigma^2}{g_{|\mathscr{V}_j|}}. \tag{12}$$

If all devices assigned to the UAV server follow the order in (12), the SIC process is successfully completed. When the order of these devices follows this decoding order, the data rate r_i^{uav} of device u_i can be denoted by

$$r_i^{\text{uav}} = B\log_2\left(1 + \frac{p_i g_i}{\sum_{t=i+1}^{|\mathscr{V}_j|} p_t g_t + \sigma^2}\right). \tag{13}$$

In the same way, we represent r_i^{bs} as the data rate of device u_i when u_i chooses to offload data by the BS, and r_i^{bs} is also denoted by

$$r_i^{\text{bs}} = B\log_2\left(1 + \frac{p_i g_i}{\sum_{t=i+1}^{|\mathscr{S}_j|} p_t g_t + \sigma^2}\right). \tag{14}$$

Combining SIC technology and the data rate of device, the upper bound Q_i of intracell interference received by device u_i can be represented by

$$\sum_{t=i+1}^{z} p_t g_t \leq Q_i, \tag{15}$$

where z represents the total number of devices on the channel selected by device u_i.

2.3. Data Offloading Model

2.3.1. Local Computing. For each device u_i, if u_i chooses local computing to offload its data, the computing delay T_i^{loc} generated by its computation task K_i is expressed as

$$T_i^{\text{loc}} = \frac{X_i}{f_i}. \tag{16}$$

We denote ε as the energy consumption factor when computing locally; the size of which is determined by the device's own chip architecture. The energy consumption generated by the computing task K_i is expressed as

$$E_i^{\text{loc}} = \varepsilon(f_i)^2 X_i. \tag{17}$$

2.3.2. Offloading Data by BS. When u_i chooses offloading its data by BS, the delay generated by the task transmission can be expressed as

$$T_i^{\text{bs}} = \frac{D_i}{r_i^{\text{bs}}}, \tag{18}$$

and the energy consumption of the task transmission is represented by

$$E_i^{\text{bs}} = p_i T_i^{\text{bs}} = p_i \frac{D_i}{r_i^{\text{bs}}}. \tag{19}$$

2.3.3. Offloading Data by UAV. When u_i chooses offloading its data by UAV, the delay generated by the task transmission can be expressed as

$$T_i^{\text{uav}} = \frac{D_i}{r_i^{\text{uav}}}, \tag{20}$$

and the energy consumption of the task transmission is represented by

$$E_i^{\text{uav}} = p_i T_i^{\text{uav}} = p_i \frac{D_i}{r_i^{\text{uav}}}. \tag{21}$$

2.4. Problem Formulation. Through the above description, we take the energy consumption of the device as the main influence part of its cost function. Assuming that the strategies a_{-i} of other devices remain unchanged, the cost generated by strategy a_i of device u_i can be expressed as

$$C_{a_{-i}}(a_i) = \begin{cases} \lambda E_i^{\text{loc}}, & a_i = 0, \\ \lambda E_i^{\text{bs}}, & a_i = 1, \\ \lambda E_i^{\text{uav}}, & a_i = 2, \end{cases} \tag{22}$$

where λ is the varying parameter of device's cost function. When the intracell interference of device u_i reaches the upper bound Q_i, the energy consumption of the device will reach the maximum value $E_i^{\max}$. Due to the instability and weak computing power of the device, we believe that the local computing energy consumption should meet

$$E_i^{\text{loc}} \geq E_i^{\max} = p_i \frac{D_i}{B\log_2(1 + (p_i g_i / Q_i + \sigma^2))}. \tag{23}$$

In this data offloading system for NOMA-enabled urban IoT, in order to reduce costs and improve efficiency, we aim to minimize the overall cost, namely,

$$\begin{aligned} &\min \sum_{u_i \in \mathcal{U}} C_{a_{-i}}(a_i) \\ &\text{s.t.}(8), (15), (23) \end{aligned} \tag{24}$$

where (8) is the NOMA-enabled data rate constraint of device u_i, (15) is u_i's received intracell interference constraint, and (23) denotes u_i's energy consumption constraint.

3. Game-Based Data Offloading for Urban Internet of Things

3.1. Game Model for Data Offloading. In our system, each device wants to minimize its own cost, which is not only determined by the decisions it makes but also affected by the decisions of other devices. Based on the competition (each device is aimed at minimizing its own cost) among the devices in our system and the resource constraints of the whole system, this problem cannot be solved with the common methods in reality [2, 32]. Potential game, a game-theoretical tool to solve the competition between devices when there are multiple constraints, is suitable for solving competition problems and minimizing the overall cost for this system. We can formulate this model as a game model of noncooperative competition among devices, defined as

$$\mathcal{G} = \left\{ n, \{a_i\}_{u_i \in \mathcal{U}}, \left\{ C_{a_{-i}}(a_i) \right\}_{u_i \in \mathcal{U}} \right\}, \tag{25}$$

where n represents the number of all devices in this system, a_i is the data offloading strategy of device u_i, and $C_{a_{-i}}(a_i)$ denotes the cost resulted by strategy a_i when other devices' strategies are given.

In model $\mathcal{G}$, due to the principle of individual rationality of devices, each device tends to reduce its own cost when the system resources are limited, and our system will reach the NE state after multiple iterations. In this state, any device unilaterally changing its own strategy (the strategies of other devices remain unchanged) will not reduce its own cost.

Definition 2 (Nash equilibrium). In this noncooperative competition game model $\mathcal{G}$, in case there is a strategy profile $\mathbf{a}^* = \{a_1^*, a_2^*, \cdots, a_n^*\}$ for all devices, each device does not want to change its strategy to reduce the cost; it can also be expressed as

$$C_{a_{-i}^*}(a_i) \le C_{a_{-i}^*}(a_i^*), \forall u_i \in \mathcal{U}. \tag{26}$$

The Nash equilibrium (NE) solution is the combination of all strategies, i.e., the stable strategies of all game players (devices) [33]. At this point, any device will not reduce its cost by changing strategy, but the existence of NE needs to be proven. There must be the NE in potential game due to potential game's Finite Improvement Property (FIP), so we need to prove that $\mathcal{G}$ is a potential game to ensure the existence of NE in our model.

Definition 3 (potential game). $\forall a_i, a_i' \in \mathbf{a}$, given the other devices' strategies a_{-i} except u_i's strategy a_i, a game $\mathcal{G}$ is an generalized ordinal potential game if there is a function $\Phi_{a_{-i}}(a_i)$ satisfying

$$C_{a_{-i}}(a_i) > C_{a_{-i}}\left(a_i'\right) \Rightarrow \Phi_{a_{-i}}(a_i) > \Phi_{a_{-i}}\left(a_i'\right). \tag{27}$$

Following the above, we need to testify that our game model $\mathcal{G}$ is a potential game through the potential game theorem, i.e., satisfying Definition 2. Before introducing this theorem, we need to define a conditional judgment variable $I_{\{\cdots\}} = \{0, 1\}$. Taking $I_{\{a_t=a_i\}}$ as an example, $I_{\{a_t=a_i\}} = 1$ when device u_t's strategy a_t and device u_i's strategy a_i are the same, and $I_{\{a_t=a_i\}} = 0$ when a_t is different from a_i [23].

Theorem 4. *$\mathcal{G}$ is a generalized ordinal potential game; its potential function $\Phi_{a_{-i}}(a_i)$ is expressed as*

$$\begin{aligned} \Phi_{a_{-i}}(a_i) = &- \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t=a_i\}}} I_{\{a_i=1\}} \\ &- \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l=a_i\}}} I_{\{a_i=2\}} - \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{Q_i} I_{\{a_i=0\}}. \end{aligned} \tag{28}$$

Proof. If we assume $C_{a_{-i}}(a_i) > C_{a_{-i}}(a_i')$, the possible strategies of device u_i are divided into four cases: Case 1: $a_i = 0, a_i' = 1$; Case 2: $a_i = 0, a_i' = 2$; Case 3: $a_i = 1, a_i' = 2$; Case 4: $a_i = 2, a_i' = 1$. □

Case 1.

$$C_{a_{-i}}(a_i) > C_{a_{-i}}\left(a_i'\right) \Rightarrow E_i^{\text{loc}} > E_i^{\text{bs}}. \tag{29}$$

According to equation (23), we can get

$$\begin{aligned} E_i^{\text{loc}} \ge E_i^{\max} \ge E_i^{\text{bs}} &\Rightarrow p_i \frac{D_i}{B\log_2(1 + (p_i g_i / Q_i + \sigma^2))} \\ &> p_i \frac{D_i}{r_i^{\text{bs}}} \Rightarrow \log_2\left(1 + \frac{p_i g_i}{Q_i + \sigma^2}\right) \\ &< \log_2\left(1 + \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t=a_i\}} + \sigma^2}\right) \Rightarrow \frac{p_i g_i}{Q_i + \sigma^2} \\ &< \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t=a_i\}} + \sigma^2} \Rightarrow \frac{p_i g_i}{Q_i} < \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t=a_i\}}}. \end{aligned} \tag{30}$$

The potential function's difference between a_i and a_i' is

$$\Phi_{a_{-i}}(a_i) - \Phi_{a_{-i}}\left(a_i'\right) = \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t = a_i\}}} - \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{Q_i} = \sum_{u_i \in \mathcal{U}} \left(\frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t = a_i\}}} - \frac{p_i g_i}{Q_i} \right) > 0. \tag{31}$$

Case 2.

$$C_{a_{-i}}(a_i) > C_{a_{-i}}\left(a_i'\right) \Rightarrow E_i^{\text{loc}} > E_i^{\text{uav}}. \tag{32}$$

Similar to Case 1, it is easy to know

$$\begin{aligned} E_i^{\text{loc}} \geq E_i^{\max} \geq E_i^{\text{uav}} &\Rightarrow p_i \frac{D_i}{B \log_2(1 + (p_i g_i / Q_i + \sigma^2))} \\ &> p_i \frac{D_i}{r_i^{\text{uav}}} \Rightarrow log_2\left(1 + \frac{p_i g_i}{Q_i + \sigma^2}\right) \\ &< \log_2\left(1 + \frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l = a_i\}} + \sigma^2}\right) \Rightarrow \frac{p_i g_i}{Q_i} \\ &< \frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l = a_i\}}}. \end{aligned} \tag{33}$$

The potential function's difference between a_i and a_i' is

$$\Phi_{a_{-i}}(a_i) - \Phi_{a_{-i}}\left(a_i'\right) = \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l = a_i\}}} - \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{Q_i} = \sum_{u_i \in \mathcal{U}} \left(\frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{S}_j|} p_l g_l I_{\{a_l = a_i\}}} - \frac{p_i g_i}{Q_i} \right) > 0. \tag{34}$$

Case 3.

$$\begin{aligned} C_{a_{-i}}(a_i) > C_{a_{-i}}\left(a_i'\right) &\Rightarrow E_i^{\text{bs}} > E_i^{\text{uav}} \Rightarrow p_i \frac{D_i}{r_i^{\text{bs}}} \\ &> p_i \frac{D_i}{r_i^{\text{uav}}} \Rightarrow \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t = a_i\}}} < \frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l = a_i\}}}. \end{aligned} \tag{35}$$

The potential function's difference between a_i and a_i' is

$$\Phi_{a_{-i}}(a_i) - \Phi_{a_{-i}}\left(a_i'\right) = \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t = a_i\}}} - \sum_{u_i \in \mathcal{U}} \frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l = a_i\}}} = \sum_{u_i \in \mathcal{U}} \left(\frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t = a_i\}}} - \frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l = a_i\}}} \right) > 0. \tag{36}$$

Case 4.

$$\begin{aligned} C_{a_{-i}}(a_i) > C_{a_{-i}}\left(a_i'\right) &\Rightarrow E_i^{\text{uav}} > E_i^{\text{bs}} \Rightarrow p_i \frac{D_i}{r_i^{\text{uav}}} \\ &> p_i \frac{D_i}{r_i^{\text{bs}}} \Rightarrow \frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l = a_i\}}} < \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t = a_i\}}}. \end{aligned} \tag{37}$$

The potential function's difference between a_i and a_i' is

$$\Phi_{a_{-i}}(a_i) - \Phi_{a_{-i}}\left(a_i'\right) = \sum_{u_i \in \mathcal{U}} \left(\frac{p_i g_i}{\sum_{l=i+1}^{|\mathcal{V}_j|} p_l g_l I_{\{a_l = a_i\}}} - \frac{p_i g_i}{\sum_{t=i+1}^{|\mathcal{S}_j|} p_t g_t I_{\{a_t = a_i\}}} \right) > 0. \tag{38}$$

According to the above description, Theorem 4 is proved.

3.2. Game-Based Offloading Algorithm Design. To solve the problem of device data offloading (which offloading method to choose), we propose a game-based data offloading algorithm (GDOA) to minimize the total cost of this system [34, 35]. In our algorithm, the necessary experimental parameters are first given, and all devices undergo a limited number of iterations, in which each device updates its strategy based on its own minimum cost, until no device is willing to update its own strategy to change the cost, i.e., the NE state is reached [36]. Our proposed algorithm is shown in Algorithm 1, and the main steps are as follows:

(1) Environment initialization: given the required experimental parameters, we initialize the strategy a_i of each device u_i to $a_i = -1$

(2) Iterative process: as shown on lines 3-14, in each iteration, we first need to calculate the cost of the system in the current state and then formulate the strategy for each device $u_i \in \mathcal{U}$. Given three data offloading approaches: local computing, base station data offloading, and UAV data offloading, before updating each device's strategy, we need to calculate the total system cost caused by these possible cases (assuming this device's strategy is one of them) and find the lowest total cost for device u_i. Then, we find the lowest cost device among all devices and update the corresponding strategy for that device

(3) End of iteration: no device is willing to update its own strategy and finally returns the strategy profile of all devices

4. Performance Evaluation

4.1. Experimental Settings. In order to correctly evaluate the simulation performance of the model, we need to give the important parameters involved in the experiment, as shown in Table 2. Besides, the coordinates of all devices need to be

Input: $\mathcal{U}, \mathcal{W}, \mathcal{V}, w_j$ and other parameters
Output: offloading strategy profile $\mathbf{a} = \{a_1, a_2, \cdots, a_n\}$
Initialization: each device u_i's strategy is $a_i = -1$
End Initialization
repeat
  calculate the current total cost $\sum_{u_i \in \mathcal{U}} C_{a_{-i}}(a_i)$
  for *each cluster* $W_j \in \mathcal{U}$ **do**
    for *each device* $u_i \in \mathcal{W}_j$ **do**
      calculate the total cost $\sum_{u_i \in \mathcal{U}} C_{a_{-i}}(a_i^1)$ when assuming $a_i^1 = 0$;
      calculate the total cost $\sum_{u_i \in \mathcal{U}} C_{a_{-i}}(a_i^2)$ when assuming $a_i^2 = 0$;
      calculate the total cost $\sum_{u_i \in \mathcal{U}} C_{a_{-i}}(a_i^3)$ when assuming $a_i^3 = 0$;
      find the minimum cost $\sum_{u_i \in \mathcal{U}} C_{a_{-i}}(a_i')$ from the three costs;
      if $\sum_{u_i \in \mathcal{U}} C_{a_{-i}}(a_i') < \sum_{u_i \in \mathcal{U}} C_{a_{-i}}(a_i)$ **then**
        compete for the opportunity to update strategy;
        if *device* u_i *wins the competition* **then**
          update its strategy from a_i to a_i';
until *there is no device willing to update its strategy*;
Return a

Algorithm 1: Game-based data offloading algorithm (GDOA).

Table 2: Experiment settings.

Parameters	Value
The number of clusters	5
The number of urban IoT devices in each cluster	10
The bandwidth of the channel	20 MHz
The transmit power of each urban IoT device	1 W
The data size of each computation task	3000 KB
The background noise	-100 dBm

given, and the channel gain between the device and the BS or UAV can be calculated [37].

4.2. Parameter Analysis. In this section, we will carry out a series of parameter analysis experiments to evaluate the performance of our model, i.e., setting different parameters to obtain the experimental results.

As shown in Figure 2, it describes the effect of different values of channel fading factor on the system cost, and the independent variable is the number of devices in each cluster. It is obvious that the system cost (cost of all devices) increases as the number of devices in each cluster increases, regardless of the channel fading factor. The increase of devices in each cluster will lead to greater interference among all devices, and the sum of the costs of all devices will definitely increase when the system resources remain unchanged. When the number of devices in each cluster is determined, the channel fading factor increases from 3 to 5, and the average data rate decreases with it. In other words, as the channel fading factor becomes larger, the interference received by the device during the data transmission process is reduced, and the total system cost is reduced.

Figure 3 shows the effect of different channel fading factors on the average data rate of all devices, where the independent variable of the experiment is the number of devices in each cluster. Regardless of the value of the channel fading factor, as the number of devices in each cluster increases, the average data rate of all devices decreases. The increase in the number of devices in each cluster will lead to increased interference, and by Shannon's theorem, the average data rate of all devices will decrease. When the number of devices in each cluster is determined, the channel fading factor increases from 3 to 5, and the average data rate decreases due to the smaller signal-to-noise ratio.

Figure 4 denotes the effect of different values of the channel fading factor on the system cost, and the independent variable is the height of the UAV in each cluster. Regardless of the channel fading factor, the system cost increases as the height of the UAV in each cluster increases. The increase in the height of the drone will result in increased interference between all devices and an overall increase in cost. When the UAV height of each cluster is determined, the channel fading factor is increased from 3 to 5, and the system cost is reduced due to the smaller signal-to-noise ratio.

Figure 5 represents the relationship between different channel fading factors and the average data rate of all devices, with the independent variable being the height of the UAV in each cluster. Obviously, the average data rate decreases as the UAV's altitude increases, because the increased altitude causes greater interference between all devices, which also affects the average data rate. And when the UAV's height of each cluster is determined, if the channel fading factor increases from 3 to 5, the average data rate decreases due to the increase of the overall interference.

The influence of different values of the channel fading factor on the data offloading ratio of the device can be obtained intuitively from Figure 6. With the increase of the channel fading factor, the proportion of the number of devices offloading locally to the total number of devices gradually decreases to zero, the number of devices offloading by the BS to the total number of devices gradually increases, and the proportion of the number of devices offloading by the UAV to the total number of devices remains unchanged. The reason is that after the channel fading factor is increased to 4, the channel gain of UAV offloading has nothing to do with the channel fading factor, and the cost of selecting BS offloading for any device is significantly lower than selecting local computing.

4.3. Comparison Experiments. In order to demonstrate the superiority of Algorithm 1 in both simulated data and the real world, our algorithm are supposed to be compared with a benchmark algorithm and several advanced algorithms. These algorithms are as follows:

(i) Random: in this algorithm, each device randomly selects an offloading way (local computing, by the

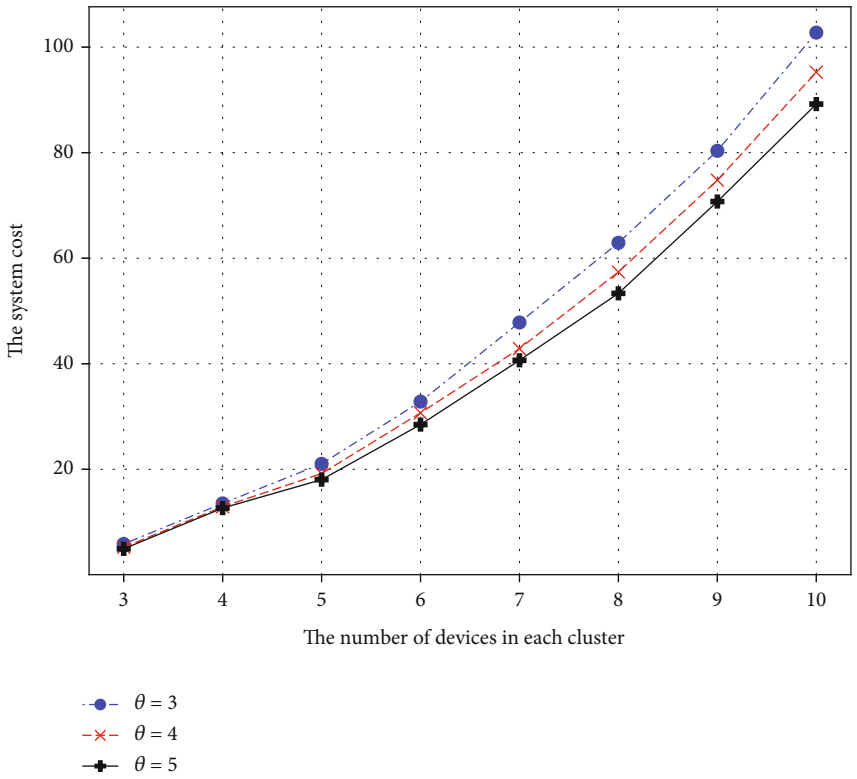

Figure 2: Effect of channel fading factor on system cost.

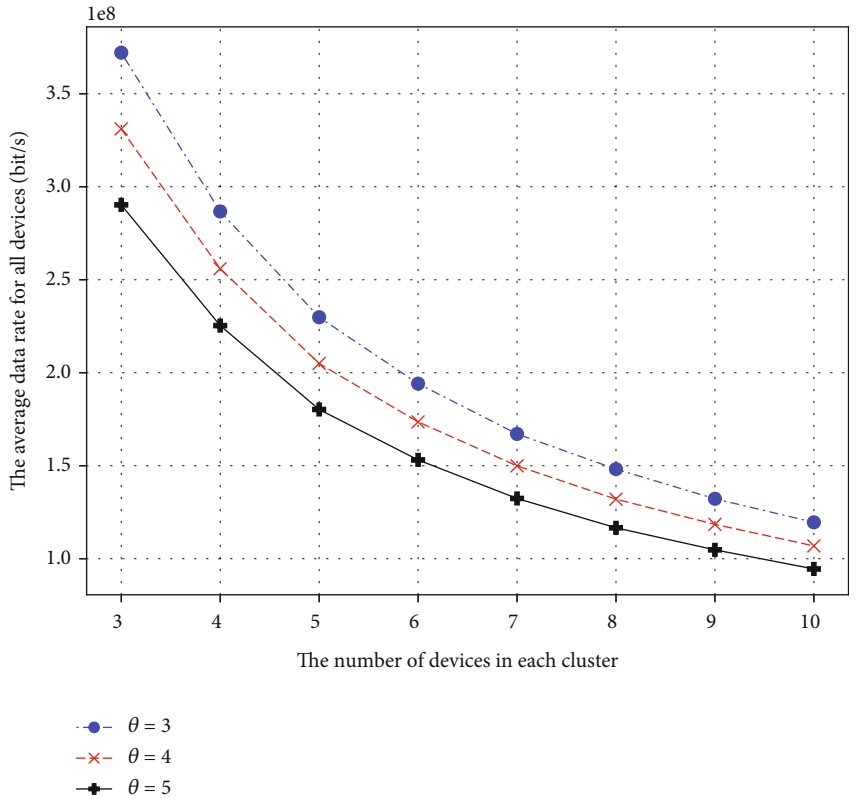

Figure 3: Effect of channel fading factor on average data rate.

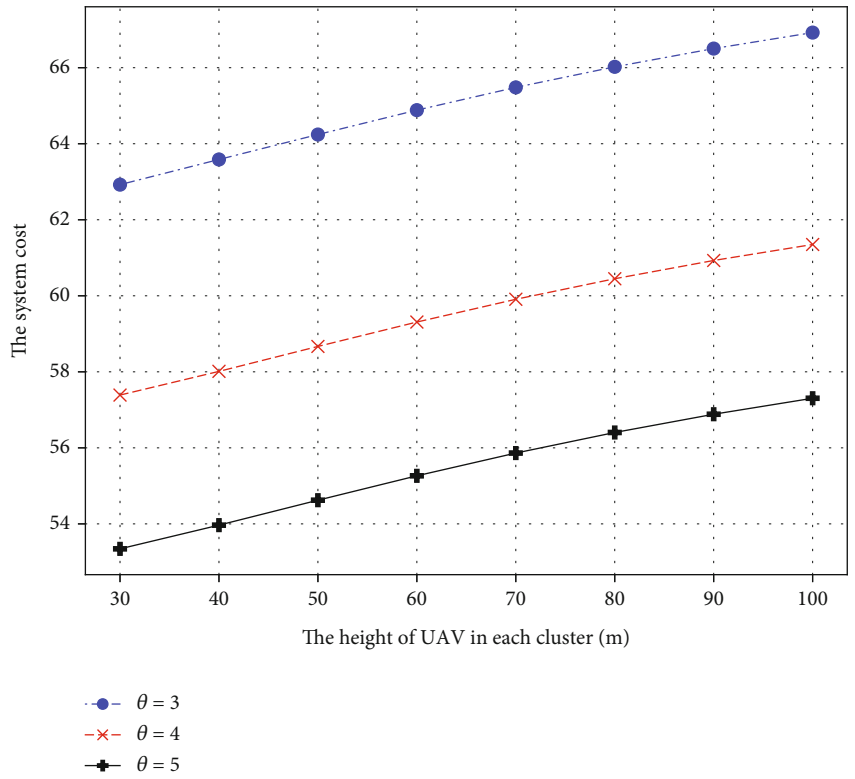

Figure 4: Effect of channel fading factor on system cost.

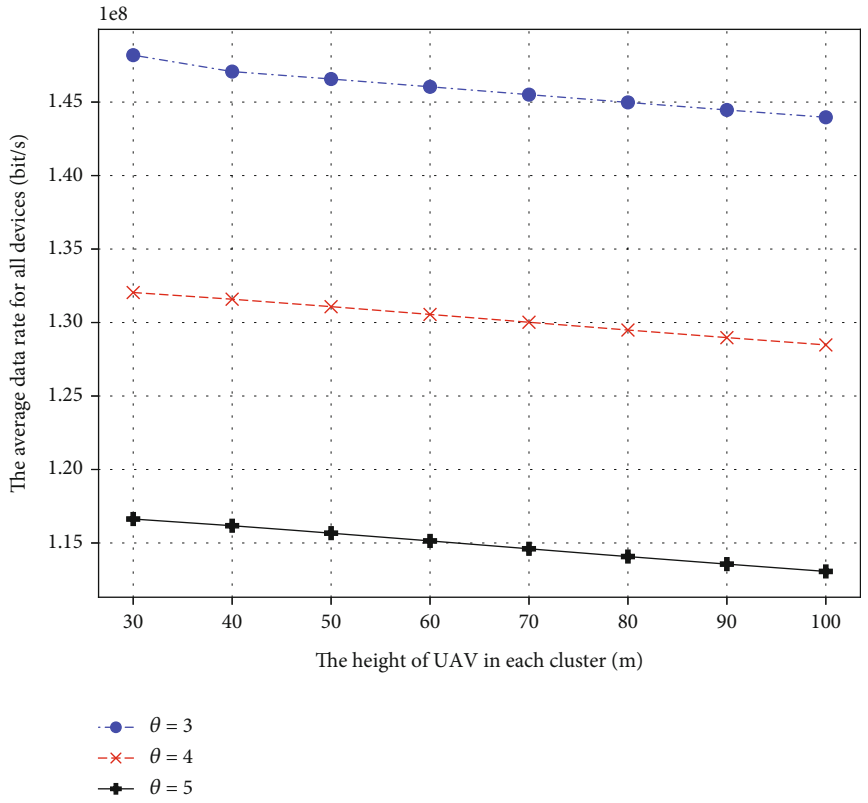

Figure 5: Effect of channel fading factor on average data rate.

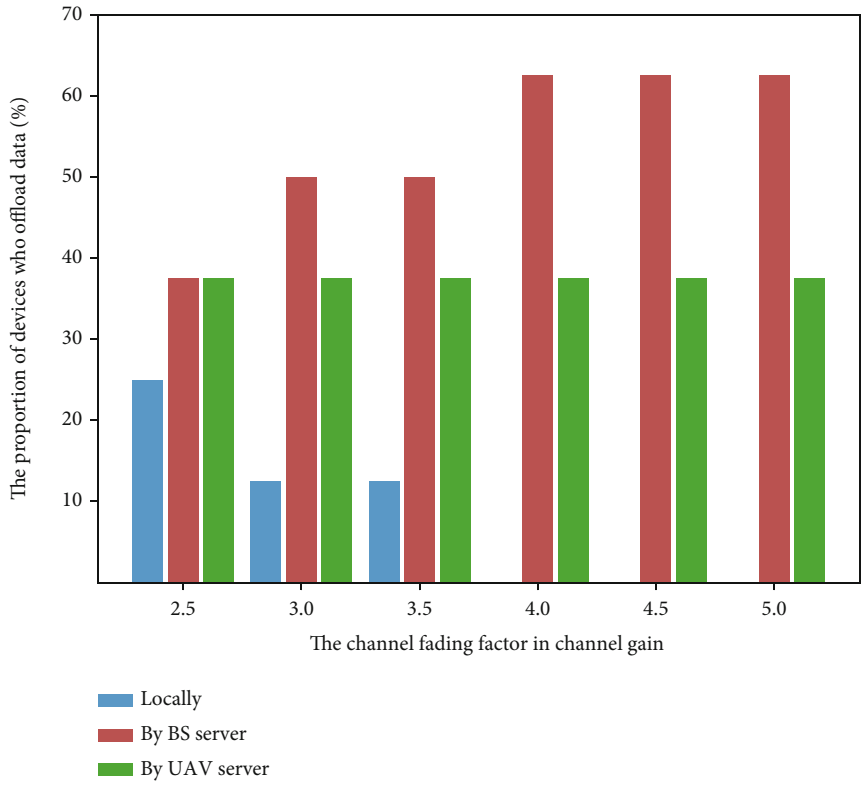

Figure 6: Effect of channel fading factor on offloading proportion.

BS or by the UAV) as its strategy and then calculates the total cost of all devices. After repeating the above process hundreds of times, the average state is the final state

(ii) OMA-DOA: a data offloading algorithm based on OMA, i.e., the calculation of intracell interference is different from NOMA. Similar to Algorithm 1, the idea of OMA-DOA is also an iterative process for each device to select the strategy that leads to the lowest system cost

(iii) ICSOC19 [38]: regardless of the minimization of system cost, each device chooses the current offloading strategy that can bring the lowest cost in each iteration

As shown in Figure 7, what can be known is the effect of the number of devices in each cluster on the system cost in different algorithms. As the number of devices increases, the system cost in all four algorithms increases. When the number of devices in each cluster increases from 3 to 5, the system cost of Random algorithm is the highest, the system cost of OMA-DOA algorithm is much higher than that of ICSOC19 algorithm, and the system cost of GDOA is the lowest. The reason is that the number of devices is not enough, and the interference in the OMA-DOA algorithm is not large enough, so the unstable Random algorithm has the highest system cost, and the GDOA is better than the local optimum of ICSOC19 algorithm due to its iterative update (reaching the global optimum). When the number of devices in each cluster increases from 6 to 8, the system cost of the OMA-DOA algorithm is the highest, the system cost of Random algorithm is much higher than that of ICSOC19 algorithm, and the system cost of the GDOA is the lowest. Similar to when the number of devices increases from 6 to 8, the gradient remains the same, when the number of devices per cluster increases from 8 to 10, but the cost gap between the algorithms becomes larger. The cause is that the increase in the number of devices leads to more and more interference in OMA-DOA algorithm, and the system cost caused by GDOA that is aimed at the overall optimal is the smallest.

What can be known in Figure 8 is the influence of the channel fading factor on the system cost in different algorithms. As the channel fading factor increases, the system cost in all four algorithms decreases. Among them, the decreasing trend of system cost of OMA-DOA algorithm is not obvious, and the decreasing trend of system cost of GDOA is the fastest. When the value of the channel fading factor is determined, the system cost of OMA-DOA algorithm is the highest, the system cost of Random algorithm is higher than that of ICSOC19 algorithm, and the system cost of GDOA is the lowest. The reason is that the larger channel fading factor will basically not affect the interference received by the equipment in OMA-DOA algorithm, which is of great help to GDOA and ICSOC19 algorithms based on SIC technology. And the iterative update property of GDOA (reaching the global optimum) is better than the local optimum of ICSOC19 algorithm.

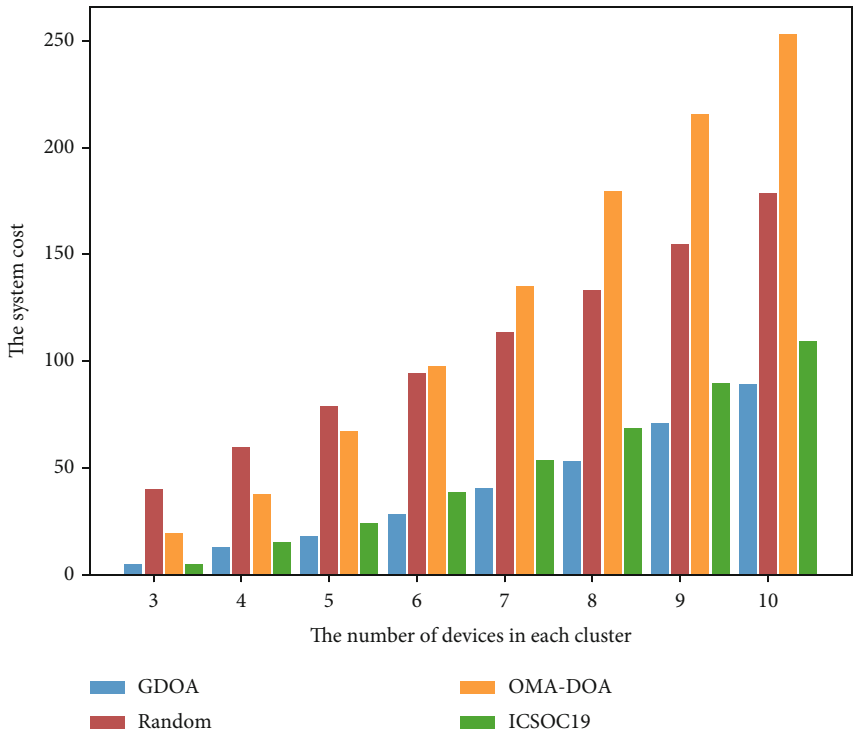

Figure 7: Effect of the number of devices on system cost.

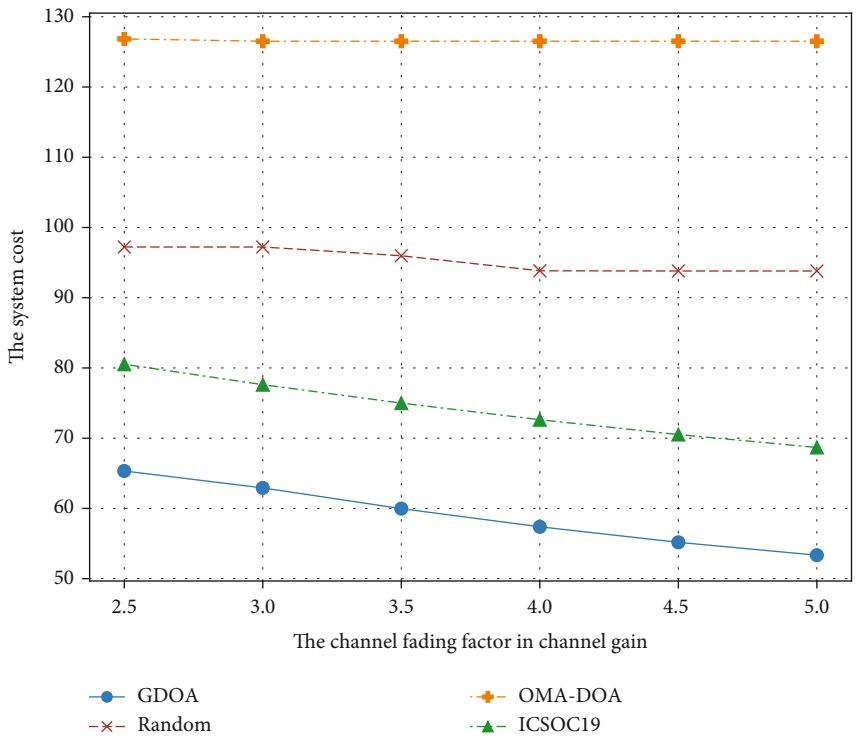

Figure 8: Effect of the channel fading factor on system cost.

5. Related Work

There is an increasing amount of work on data offloading in Internet of Things systems. The authors of [39] studied the problem of computation offloading based on age of information, while considering the energy control of its system. But this paper did not consider the complex data access of multiple BSs. In [40], the authors proposed an intelligent video application based on interference-aware data offloading, which can be combined with real-time traffic monitoring

to improve urban IoT. The authors in [41] considered the energy-saving task offloading problem based on stochastic optimization in MEC, which minimizes the energy consumption of task offloading while ensuring the average queue length. Reference [42] proposed an edge computing solution that can achieve mutual cooperation between different edge servers in the case of limited resources, thereby effectively reducing the delay of data delivery. However, data offloading or edge computing in these papers does not consider using UAVs to assist BSs for data services.

In order to solve the problem of intermittent network connection of mobile devices, many work has introduced UAV-assisted BS to provide devices with seamless coverage of urban IoT services for the increasingly complex requirements. In [14], the authors aimed to jointly optimize the coverage radius of UAV servers and the number of UAV servers to maximize the minimum throughput of ground users at the cell edge through nonorthogonal spectrum reuse. And [43] studied the need for IoT devices to offload some tasks to the edge server of the base station and the drone server with controllable maneuverability and flexibility under the premise of harsh environment. Nevertheless, data offloading or edge computing in these papers does not consider nonorthogonal multiple access during data transmission.

In recent years, NOMA technology in 5G IoT has been plenty of studied to improve the offloading performance and reduce the system cost. Reference [44] proposed a novel framework named Edge Game, which effectively improves the user's data rate and service experience by using NOMA to remove the successive interference generated when offloading computationally intensive tasks. In [45], the authors studied the NOMA-supported energy-saving task offloading and resource allocation problems in the intelligent IoT, solved the limitation of computing and storage resources through NOMA technology, and used advanced stochastic optimization technology to minimize energy consumption while satisfying delay constraints. However, considering the limited computing resources of the system and the inevitability of intracell interference, it is obviously unrealistic to ignore the influence of device transmit power on the SIC process.

In this paper, we study the decision problem of offloading approach combining data offloading and device allocation. We minimize the total energy consumption of urban IoT devices through stochastic optimization, transform it into a potential game model by applying potential game theory, and propose GDOA to verify the excellent performance of this model. In addition, we also introduce NOMA technology to cancel the successive intracell interference.

6. Conclusion

This paper studies the NOMA-based device data offloading in an urban IoT system, and the BS and UAVs in the urban IoT system provide data offloading services to the devices. We adopt SIC technology to effectively eliminate the interference between devices when offloading data so that each device can obtain a reasonable data rate while satisfying the SIC ordering. Our optimization goal is to minimize the energy cost of all equipment. Since this offloading problem is difficult to solve in the real world, we define a potential game model of noncooperative competition among devices. And we propose an iterative update-based GDOA for this game model, which can reach the Nash equilibrium state with the minimum system cost at the end of the iteration. Then, we use the simulation data to conduct parameter analysis experiments and comparison experiments to verify the superiority of our proposed GDOA in real-world applications. In future work, we will consider cooperative data offloading between dynamic UAVs and base stations, which may involve graph optimization and dynamic game theory.

Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

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

References

[1] S. Ali, M. Sohail, S. B. H. Shah et al., "New trends and advancement in next generation mobile wireless communication (6g): a survey," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 9614520, 14 pages, 2021.

[2] A. D. Boursianis, M. S. Papadopoulou, A. Gotsis et al., "Smart irrigation system for precision agriculture—the arethou5a iot platform," *IEEE Sensors Journal*, vol. 21, no. 16, pp. 17539–17547, 2020.

[3] P. Wang, C. Lin, M. S. Obaidat, Z. Yu, Z. Wei, and Q. Zhang, "Contact tracing incentive for covid-19 and other pandemic diseases from a crowdsourcing perspective," *IEEE Internet of Things Journal*, vol. 8, no. 21, pp. 15863–15874, 2021.

[4] C. Chen, L. Liu, S. Wan, X. Hui, and Q. Pei, "Data dissemination for industry 4.0 applications in Internet of vehicles based on short-term traffic prediction," *ACM Transactions on Internet Technology (TOIT)*, vol. 22, no. 1, pp. 1–18, 2022.

[5] A. Hudson-Smith, D. Wilson, S. Gray, and O. Dawkins, "Urban iot: advances, challenges, and opportunities for mass data collection, analysis, and visualization," in *Urban Informatics*, pp. 701–719, Springer, 2021.

[6] D. Amaxilatis, D. Boldt, J. Choque et al., "Advancing experimentation-as-a-service through urban iot experiments," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2563–2572, 2018.

[7] F. Rebecchi, M. D. De Amorim, V. Conan, A. Passarella, R. Bruno, and M. Conti, "Data offloading techniques in cellular networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 580–603, 2015.

[8] H. Zhou, H. Wang, X. Li, and V. C. Leung, "A survey on mobile data offloading technologies," *IEEE access*, vol. 6, pp. 5101–5111, 2018.

[9] P. Wang, R. Yu, N. Gao, C. Lin, and Y. Liu, "Task-driven data offloading for fog-enabled urban iot services," *IEEE Internet of Things Journal*, vol. 8, no. 9, pp. 7562–7574, 2020.

[10] D. Mazza, D. Tarchi, and G. E. Corazza, "A unified urban mobile cloud computing offloading mechanism for smart cities," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 30–37, 2017.

[11] P. Wang, Y. Pan, C. Lin et al., "Graph optimized data offloading for crowd-ai hybrid urban tracking in intelligent transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–13, 2022.

[12] P. Kortoçi, L. Zheng, C. Joe-Wong, M. Di Francesco, and M. Chiang, "Fog-based data offloading in urban iot scenarios," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 784–792, Paris, France, May 2019.

[13] S. Guntuka, E. Shakshuki, and A. Yasar, "Iot mobile device data offloading by small-base station using intelligent software defined network," *Procedia Computer Science*, vol. 177, pp. 234–244, 2020.

[14] Y. Chen, F. Zhao, Y. Lu, and X. Chen, "Dynamic task offloading for mobile edge computing with hybrid energy supply," *Tsinghua Science and Technology*, vol. 10, 2021.

[15] H. Wang, H. Wang, and J. An, "Dynamic game-based computation offloading and resource allocation in LEO- multiaccess edge computing," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 9486818, 13 pages, 2021.

[16] 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*, p. e5161, 2022.

[17] Y. Chen, H. Xing, Z. Ma, X. Chen, and J. Huang, "Cost-efficient edge caching for noma-enabled iot services," *China Communications*, 2022.

[18] T. Rahman, F. Khan, I. Khan, N. Ullah, M. M. Althobaiti, and F. Alassery, "NOMA and oma-based massive mimo and clustering algorithms for beyond 5g iot networks," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6522089, 12 pages, 2021.

[19] L. Dai, B. Wang, Z. Ding, Z. Wang, S. Chen, and L. Hanzo, "A survey of non-orthogonal multiple access for 5g," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2294–2323, 2018.

[20] W. Wu, F. Zhou, R. Q. Hu, and B. Wang, "Energyefficient resource allocation for secure Noma-enabled mobile edge computing networks," *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 493–505, 2020.

[21] Z. Li, M. Chen, Z. Yang et al., "Energy efficient reconfigurable intelligent surface enabled mobile edge computing networks with Noma," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 2, pp. 427–440, 2021.

[22] 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, 2022.

[23] 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.

[24] T.-H. Vu, T.-V. Nguyen, and S. Kim, "Cooperative nomaenabled swipt iot networks with imperfect sic: performance analysis and deep learning evaluation," *IEEE Internet of Things Journal*, vol. 9, no. 3, pp. 2253–2266, 2021.

[25] L. P. Qian, A. Feng, Y. Huang, Y. Wu, B. Ji, and Z. Shi, "Optimal sic ordering and computation resource allocation in mecaware Noma nb-iot networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2806–2816, 2018.

[26] Y. Chen, N. Zhang, Y. Zhang, X. Chen, W. Wu, and X. S. Shen, "TOFFEE: task offloading and frequency scaling for energy efficiency of mobile devices in mobile edge computing," *IEEE Transactions on Cloud Computing*, vol. 9, no. 4, pp. 1634–1644, 2021.

[27] P. Radoglou-Grammatikis, K. Rompolos, P. Sarigiannidis et al., "Modeling, detecting, and mitigating threats against industrial healthcare systems: a combined software defined networking and reinforcement learning approach," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 2041–2052, 2021.

[28] 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.

[29] C. Chen, Y. Zhang, Z. Wang, S. Wan, and Q. Pei, "Distributed computation offloading method based on deep reinforcement learning in ICV," *Applied Soft Computing*, vol. 103, p. 107108, 2021.

[30] 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, no. 2, pp. 242–247, 2020.

[31] 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.

[32] N. Ye, X. Li, H. Yu, L. Zhao, W. Liu, and X. Hou, "DeepNOMA: a unified framework for Noma using deep multitask learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2208–2225, 2020.

[33] 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.

[34] Z. Ding, J. Xu, O. A. Dobre, and H. V. Poor, "Joint power and time allocation for NOMA–MEC offloading," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 6207–6211, 2019.

[35] Z. Kuang, L. Li, J. Gao, L. Zhao, and A. Liu, "Partial offloading scheduling and power allocation for mobile edge computing systems," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6774–6785, 2019.

[36] A. Sultana, I. Woungang, A. Anpalagan, L. Zhao, and L. Ferdouse, "Efficient resource allocation in SCMA-enabled device-to-device communication for 5g networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5343–5354, 2020.

[37] S. Zhang, J. Li, H. Luo, J. Gao, L. Zhao, and X. S. Shen, "Low-latency and fresh content provision in informationcentric vehicular networks," *IEEE Transactions on Mobile Computing*, vol. 21, no. 5, 2020.

[38] P. Lai, Q. He, G. Cui et al., "Edge user allocation with dynamic quality of service," in *International Conference on Service-*

Oriented Computing, Service-Oriented Computing, pp. 86–101, Springer, Cham, 2019.

[39] J. Huang, H. Gao, S. Wan, and Y. Chen, "AoI-aware energy control and computation offloading for industrial IoT," *Future Generation Computer Systems*, 2022.

[40] X. Zhang, Y. Zhao, G. Min, W. Miao, H. Huang, and Z. Ma, "Intelligent video ingestion for real-time traffic monitoring," *ACM Transactions on Sensor Networks (TOSN)*, vol. 18, no. 3, pp. 1–13, 2022.

[41] Y. Chen, N. Zhang, Y. Zhang, X. Chen, W. Wu, and X. Shen, "Energy efficient dynamic offloading in mobile edge computing for Internet of things," *IEEE Transactions on Cloud Computing*, vol. 9, no. 3, pp. 1050–1060, 2021.

[42] X. Zhang, Z. Qi, G. Min, W. Miao, Q. Fan, and Z. Ma, "Cooperative edge caching based on temporal convolutional networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 9, pp. 2093–2105, 2021.

[43] 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. 222, p. 109218, 2022.

[44] X. Zhang, H. Chen, Y. Zhao et al., "Improving cloud gaming experience through mobile edge computing," *IEEE Wireless Communications*, vol. 26, no. 4, pp. 178–183, 2019.

[45] K. Li, J. Zhao, J. Hu, and Y. Chen, "Dynamic energy efficient task offloading and resource allocation for NOMA- enabled IoT in smart buildings and environment," *Building and Environment*, 2022.