Journal of Computers Vol. 32 No. 5, 2021, pp. 192-202
doi:10.53106/199115992021103205016

Research on High Confidence Resource Allocation Technology in Edge Computing

Zhenjiang Zhang[1*], Quancheng Zhao[1], Xuan Zhao[1], Wei Lin[2], Shuai Wu[2]

[1] Key Laboratory of Communication and Information Systems, Beijing Municipal Commission of Education, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, 100044, China
{zhangzhenjiang,19125077}@bjtu.edu.cn; 921628810@qq.com

[2] Beijing MetarNet Technology Co., Ltd, Beijing, 100089, China
{lin.wei, wushuai}@inspur.com

Received 1 August 2021; Revised 1 September 2021; Accepted 8 October 2021

Abstract. This paper studies resource allocation and security performance. Aiming at the problem of privacy disclosure caused by eavesdropping during task unloading in single cell and multi user scenarios, the physical layer security technology is used to formulate corresponding confidentiality measures. Facing the dynamic change of the wireless channel state of the system and the computing power required by users, taking reducing the average processing delay of tasks as the optimization goal, the multi-user task partial migration problem is modeled as a joint optimization problem of computing resources and power resources under the constraints of security set-tings and energy constraints, and the system is established as a Markov decision process model, and proposes a resource allocation algorithm based on physical layer security and depth deterministic policy gradient. Simulation results show that the algorithm can approach the optimal performance, has adaptability and lower computational complexity, can make better unloading strategy and resource allocation scheme in trusted environment, effectively reduce the average task processing delay and improve the robustness of the system.

Keywords: edge computing, resource allocation, task offloading, physical layer security

1 Introduction

1.1 Research Background and Significance

In recent years, with the in-depth development of 5G communication technology, Internet of Things technology and computer science technology, many emerging technologies are rapidly pushing human society into the era of Internet of Everything, but what followed was the explosive growth of data traffic and the congested link load of the mobile core network [1]. The traditional cloud computing centralized data processing architecture faces various disadvantages: such as real-time issues [2], energy consumption issues, and network bandwidth issues. In order to solve the above problems, edge computing and cloud edge integration architecture are widely used. Edge computing is a distributed computing model that performs data processing and analysis on the edge of the network. It has the following advantages: it can meet the requirements of real-time data processing and low delay response [3]; it can effectively alleviate the occupation of network bandwidth; it can greatly reduce energy consumption. Although edge computing has many significant advantages, it also faces problems such as uneven resource allocation, insecure data transmission and distrust between entities. In the face of possible security issues such as privacy data eavesdropping, malicious attacks between devices, as well as the complex and changeable status of network resources, while discussing the construction of highly trusted edge computing environment, how to design reasonable and efficient task migration and resource allocation strategy has very important research value and practical significance.

* Corresponding Author

1.2 Research Status at Home and Abroad

In recent years, edge computing has become a hot research field. In foreign countries, Nokia and IBM jointly launched an edge computing platform for 4G/LTE networks in early 2013, called radio application cloud server [4]. At home, China Academy of information technology, Huawei, arm and other units jointly founded the first edge computing industry alliance in 2016, and successively released application-oriented white papers in the field of edge computing [5]. In the industry, Microsoft launched Azure Edge Zones platform for 5G and edge computing, which is mainly used in the fields of Internet of Things, AI and real-time analysis [6]. In academia, Shi et al. first pointed out the inherent problems of cloud computing mode, then proposed the definition and principle of edge computing, and instantiated the concept of edge computing in practical scenarios such as task migration, intelligent transportation and cloud edge collaboration [7]. Peng et al. described the definition, architecture, service and other basic concepts of mobile edge computing in detail, and gave a comprehensive over-view of its unloading scheme and application scenario [8]. Hassan et al. mainly introduced the advantages and performance indicators of the combination scheme of 5G network and edge computing, as well as the actual application scenarios and deployment requirements. Finally, they described the problems and challenges brought by the integration of edge computing and 5G [9]. Wang et al. proposed an intelligent resource allocation scheme based on deep Q network algorithm to obtain the optimal average service delay and load balancing effect [10]. Huang et al. studied the security problems in the scenario of the combination of vehicle networking and edge computing, and proposed a distributed reputation management mechanism to ensure the network security in vehicle edge computing [11]. Wang et al. studied the edge computing environment with multiple mobile devices and malicious eavesdroppers, adopted the physical layer security technology to ensure the safe transmission of task unloading process, and jointly optimized the local computing frequency, upload power, transmission delay and task allocation strategy based on the convex difference algorithm to reduce the total energy consumption of the system [12].

1.3 Research Content and Main Work

This paper focuses on the resource allocation and security performance in the edge computing scenario. For the multi-user and single cell scenario, the corresponding task unloading and resource allocation algorithms in the highly trusted environment are proposed to meet the needs of low delay and low energy consumption of the overall user terminal, and improve the computing service performance and balanced utilization of resources. The specific research contents of this paper are as follows:

(1) This paper expounds the concept and system architecture of edge computing, introduces the relevant theories of physical layer security technology in detail, and finally introduces the basic theories related to reinforcement learning.

(2) Aiming at the problem of privacy disclosure caused by eavesdropping during task unloading in single cell and multi-user scenarios, the physical layer security technology is adopted to formulate corresponding confidentiality measures, and the problem of partial task unloading of multiple user terminals is transformed into the problem of joint allocation of computing resources and power resources under security settings and energy constraints. According to the system scenario, the Markov decision process model is established, and a resource allocation algorithm based on physical layer security and depth deterministic policy gradient is proposed. Finally, the PyTorch framework is used for experimental simulation. The simulation results show that the algorithm has lower computational complexity than the traditional algorithm. Facing the dynamic environment, it can quickly make task unloading and resource allocation strategies, effectively reduce the average processing delay of tasks, and improve the user experience and system stability.

2 The Theoretical Basis of Edge Computing Resource Allocation and Security Mechanism

2.1 Edge Computing Overview

Edge computing is an extension of cloud computing. By “sinking” the cloud function to the edge close to the data source, it can meet the key needs of the digital industry for agile connection, data processing, real-time services and security protection. It is the best solution to effectively reduce delay and energy

consumption, improve security performance and alleviate bandwidth occupation based on the Internet of things scenario. Edge computing paradigm has been widely used in smart home, health care, smart grid, environmental intelligent monitoring, tactile Internet and other new fields [13-14].

At present, researchers have proposed several edge computing architectures for different application scenarios. The focus is to solve the limitations of taking the cloud center as the backbone architecture, such as the edge computing reference architecture shown in Fig. 1 below. The architecture includes terminal device layer, edge computing layer and cloud computing layer.

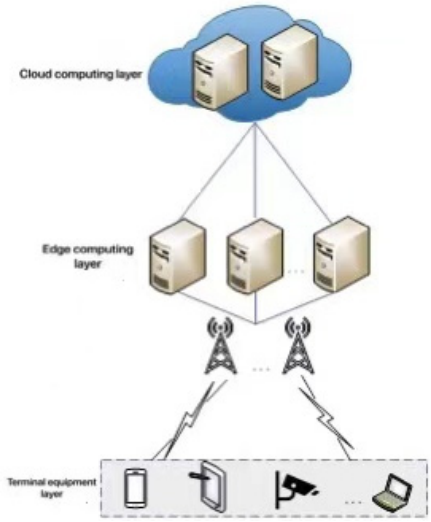

Fig. 1. Edge computing architecture

2.2 Theoretical Basis of Physical Layer Security

The physical layer security technology is based on information theory. Its basic idea is to realize the secure transmission of information by using the difference between the main channel and the eavesdropping channel (Fig. 2) due to the randomness of the wireless channel without the help of the key [15]. It can be used as a supplementary mechanism for the upper layer encryption measures to improve the security performance of the wireless communication system. It does not depend on the computing power of user terminal hardware facilities, so it is a lightweight security measure.

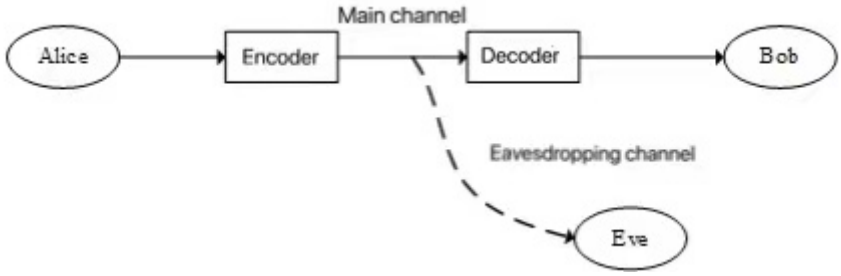

Fig. 2. Eavesdropping channel model

2.3 Reinforcement Learning Overview

Reinforcement learning is a new technology in the field of machine learning. It can give an accurate solution to the decision-making problem in complex and changeable environment. It is necessary to adjust the behavior strategy by delaying the reward value to make the model approach the correct goal. The exploration and utilization mechanism is the core mechanism of reinforcement learning. In the basic framework of reinforcement learning based on the Markov decision process, there is an agent that obtains the current system state by observing the surrounding environment S_t, after taking a certain behavior or

action A_t according to the current strategy, it enters the new system state S_{t+1}, and at the same time obtains the environmental reward or punishment feedback R_t.

3 Resource Allocation Method Based on Physical Layer Security and Depth Deterministic Policy Gradient

3.1 System Model

As shown in Fig. 3 below, in the single cell, multi-user and edge computing scenario with a malicious eavesdropper, multiple user terminals UT are connected to the base station BS, and the BS is connected to the edge server ECS through optical fiber. At the same time, the base station can be divided into multiple separate sub-channels for different user terminals UT.

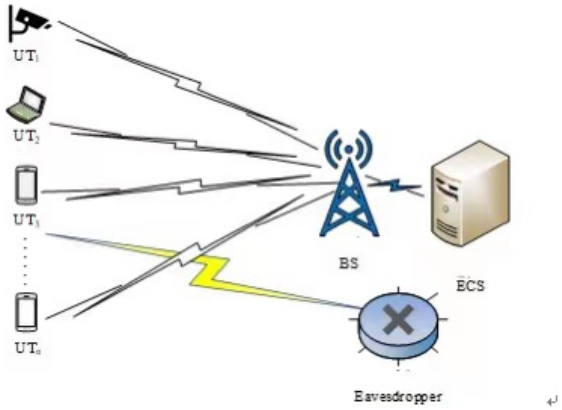

Fig. 3. Single cell multiple users and a malicious eavesdropper in edge computing

Suppose that the user terminal is represented by $N=\{1,2,\ldots,n,...,N\}$, where n is the number of UT, UT communicates with the base station BS through the wireless channel, and ECS is the edge computing server. The task model is represented by $T_n=(S_n,M_n)$, where S_n represents the length of computing tasks generated by the user terminal UT_n. M_n represents the number of CPU cycles required to process each bit of data. The unloading ratio of user terminal UT_n in t time slot is defined as a_n^t $0\le a_n^t\le 1$, that is, the amount of task data $(1-a_n^t)S_n$ is processed locally, and the amount of $a_n^t S_n$ is unloaded to ECS for processing. Then the unloading decision vector of all user terminals in a single cell in the t time slot is:

$$\Lambda(t)=[\alpha_1^t,\alpha_2^t,\ldots,\alpha_n^t,...,\alpha_N^t] \tag{1}$$

This section formulates confidentiality measures based on the safety interruption probability index. The smaller the index, the more the user needs a safer task offloading environment. According to the principle of physical layer security [16], under perfect confidentiality conditions, the maximum transmission rate R_n^{sec} of the user terminal UT_n during offloading tasks to ECS is expressed as:

$$R_n^{\text{sec}}=B\left[\log_2\left(1+\frac{p_n^t g_n^t}{n_0}\right)-\log_2\left(1+\frac{p_n^t g_\varepsilon}{n_\varepsilon}\right)\right]^+ \tag{2}$$

Only when the offloading rate R_n of the user terminal UT_n does not exceed its maximum transmission rate R_n^{sec} to the base station BS, can it be ensured that a malicious attacker cannot monitor any offloading information. Define the actual task unloading rate of the user terminal UT_n as R_n, then the probability of a security interruption event during the process of UT_n unloading tasks to the edge server ECS is:

$$P_{out}(p_n^t, R_n) = \Pr\left\{ R_n \geq R_n^{\sec} \mid B\log_2\left(1+\frac{p_n^t g_n^t}{n_0}\right) - B\log_2\left(1+\frac{p_n^t g_\varepsilon}{n_\varepsilon}\right) \geq 0 \right\} \tag{3}$$

Let $\varepsilon = P_{out}(p_n^t, R_n)$ denote the probability of safe outage. After similar probability derivation and proof [16], we can get:

$$R_n = B\log_2\left(\frac{p_n^t \hat{g}_n^t + n_\varepsilon}{p_n^t \lambda + n_\varepsilon}\right) \tag{4}$$

$$\lambda = -\beta_\varepsilon \ln(1-(1-e^{-\frac{\hat{g}_n^t}{\beta_\varepsilon}})(1-\varepsilon)) \tag{5}$$

Assuming that the local computing capability of the user terminal UT_n is f_n^{loc} c, the local execution delay of UT_n can be obtained from the unloading decision variable $\Lambda(t)$ of the user terminal in the cell at the time slot *t*:

$$t_n^{loc} = \frac{(1-\alpha_n^t)S_n M_n}{f_n^{loc}} \tag{6}$$

Since the energy of the user terminal is limited, the local processing energy consumption also needs to be considered:

$$e_n^{loc} = (1-\alpha_n^t)\zeta_n f_n^{loc^2} S_n M_n \tag{7}$$

The transmission delay t_n^{up} required by UT_n in the process of unloading $\alpha_n^t S_n$ data volume to base station BS is shown in Equation (8). Due to the security setting of the system, the task unloading rate R_n of user terminal UT_n can only get the effective security part $R_n(1-\varepsilon)$:

$$t_n^{up} = \frac{\alpha_n^t S_n}{R_n(1-\varepsilon)} \tag{8}$$

The unloading energy consumption of UT_n can be recorded as:

$$e_n^{up} = p_n^t t_n^{up} = \frac{\alpha_n^t S_n p_n^t}{R_n(1-\varepsilon)} \tag{9}$$

Suppose that the edge computing server ECS adopts a first-come, first-served processing strategy to deal with computing tasks uploaded by all user terminals in time slot t. At this time, the amount of idle computing resources of ECS is f_{ECS}^t, so the task computing delay in ECS is:

$$t_n^{ECS} = \frac{\alpha_n^t S_n M_n}{f_{ECS}^t} \tag{10}$$

Assuming that the battery energy of each user terminal UT is limited and the maximum available energy is $E_{\max}$, the sum of the local energy consumption e_n^{loc} and the unloading energy consumption e_n^{up} of UT_n cannot exceed $E_{\max}$, that is, it satisfies:

$$(1-\alpha_n^t)\zeta_n f_n^{loc^2} S_n M_n + \frac{\alpha_n^t S_n p_n^t}{R_n(1-\varepsilon)} \leq E_{\max} \tag{11}$$

For each user terminal, the local and offloading calculation processes can be performed in parallel, so the total time delay t_n to complete a calculation task processing can be expressed as:

$$t_n = \max\{t_n^{loc}, t_n^{up}, t_n^{ECS}\} = \max\left\{\frac{(1-\alpha_n^t)S_n M_n}{f_n^{loc}}, \frac{\alpha_n^t S_n}{R_n(1-\varepsilon)}, \frac{\alpha_n^t S_n M_n}{F_{ECS}^t}\right\} \tag{12}$$

In view of the dynamic change of communication resources and computing power required by users, the multi-user task partial unloading problem is modeled as a joint optimization problem of computing resources and power resources under security settings and energy constraints, and the optimization goal is to minimize the average processing delay of tasks. In conclusion, the optimization problem P can be modeled as:

$$
\begin{aligned}
&P: \min_{\alpha_n^t,\, p_n^t} \frac{1}{N}\sum_{n=1}^{N} t_n \\
&s.\ t.\ C1:\ \alpha_n^t \in [0,1] \\
&\text{C2:}\ \ p_{\min} \le p_n^t \le p_{\max} \\
&\text{C3:}\ \ \sum_{n=1}^{N} \alpha_n^t S_n M_n \le F_{ECS} \\
&\text{C4:}\ \ (1-\alpha_n^t)\zeta_n f_n^{loc^2} S_n M_n + \frac{\alpha_n^t S_n p_n^t}{R_n(1-\varepsilon)} \le E_{\max}
\end{aligned}
\tag{13}
$$

Among them, the constraint item C1 requires that the task offload ratio of each user terminal must be within the interval [0, 1]; the constraint item C2 requires the power allocation decision p_n^t to meet the upper and lower limits of the power interval; the constraint item C3 requires that the computing power required for the tasks offloaded by all users in time slot t must not exceed the upper limit of computing resources F_{ECS} provided by ECS; constraint C4 requires that the sum of the local computing energy consumption and migration energy consumption of each user terminal cannot exceed the maximum available energy F_{ECS}.

3.2 Algorithm Design

The hierarchical joint optimal algorithm for solving the original problem P is shown in Table 1.

Table 1. Description of hierarchical joint optimal algorithm

Algorithm 1. Hierarchical joint optimal algorithm

1. **For** UT=1, 2, ..., n, ..., N:
2. Initialize $t_n^* = \infty$, $\alpha_n^t = 0$, $t_{avg}^* = 0$, step size Δ;
3. **For** $\alpha_n^t = 0 : \Delta : 1$:
4. Calculate the amount of computing resources required for ECS $\alpha_n^t S_n M_n$, if $\alpha_n^t S_n M_n > f_{ECS}^t$ then exit the inner loop;
5. According to the given α_n^t, respectively calculate $F'(p_{\min})$, $F'(p_{\max})$;
6. If $F'(p_{\min}) < 0$, solve $p_n^{t,*}$, otherwise go to step 7;
7. If $F'(p_{\min}) > 0$ and $F'(p_{\max}) < 0$, solve $p_n^{t,*}$, otherwise go to step 8;
8. If $F'(p_{\max}) > 0$, solve $p_n^{t,*}$;
9. Calculate $t_n^{tmp} = \max\left\{\frac{(1-\alpha_n^t)S_n M_n}{f_n^{loc}}, \frac{\alpha_n^t S_n}{R_n(1-\varepsilon)} + \frac{\alpha_n^t S_n M_n}{F_{ECS}^t}\right\}$;
10. If $p_n^{tmp} < t_n^*$, let $t_n^* = t_n^{tmp}$, $\alpha_n^{t,*} = \alpha_n^t$, and record $p_n^{t,*}$;
11. **End For**
12. Update the current free resource amount of edge server ECS f_{ECS}^t;
13. $t_{avg}^* = t_n^* / N + t_{avg}^*$;
14. **End For**
15. Output t_{avg}^* and $p_n^{t,*}$ of each UT;

The execution of the hierarchical algorithm takes a lot of time, resulting in a long waiting time for the user terminal, which seriously affects the quality of experience. In order to solve the above problems, the

RAPLSDDPG algorithm based on deep deterministic strategy and physical layer security is proposed. After the algorithm training is completed, an effective resource allocation plan can be quickly made in a dynamic environment.

$$Loss(\theta^{Q}) = \frac{1}{n}\sum_{i}(y_i - Q(s_i, \alpha_i \mid \theta^{Q}))^2 \tag{14}$$

$$\nabla(\theta^{\mu}) \approx \frac{1}{n}\sum_{i}\nabla_a Q(s, \alpha \mid \theta^{Q})|_{s=s_V \alpha=\mu(s_i)} \nabla_{\theta^{\mu}} \mu(s \mid \theta^{\mu})|_{s_i} \tag{15}$$

$$\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'} \tag{16}$$

$$\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'} \tag{17}$$

$$0 \le \alpha_n^t \le \min\left\{\frac{E_{\max} - \zeta_n f_n^{loc^2} S_n M_n}{\frac{S_n p_n^t}{R_n(1-\varepsilon)} - \zeta_n f_n^{loc^2} S_n M_n}, 1\right\} \tag{18}$$

The following is the specific process of the RAPLSDDPG algorithm (Table 2):

Table 2. RAPLSDDPG algorithm description

Algorithm 1. RAPLSDDPG algorithm	
1.	Initialize the experience pool R and the probability of safe outage ε;
2.	Randomly initialize the Actor network $\mu(s \mid \theta^{\mu})$ and Critic network $q(s, \alpha \mid \theta^{Q})$ and their respective weight parameters;
3.	Initialize the corresponding Target network μ' and Q', and the weight parameters are $\theta^{\mu'} = \theta^{\mu}$, $\theta^{Q} = \theta^{Q}$;
4.	**For** episode =1 to max_ episodes:
5.	Initialize the noise random variable ψ to obtain the initial state of the system s_0;
6.	**For** t=1 to max_ steps:
7.	Output actions according to the current strategy network and noise disturbance and (18) restriction conditions: $\alpha_t = \mu(s_t \mid \theta^{\mu}) + \psi$;
8.	Perform the action α_t to interact with the edge computing environment to get the reward r_t, and the next state s_{t+1};
9.	Store the sample data $(s_t, \alpha_t, r_t, s_{t+1})$ into the experience pool R;
10.	Random sampling Z sample data in experience pool R $(s_i, \alpha_i, r_i, s_{i+1})$ constitute mini-batch;
11.	Calculate $y_i = r_i + rQ'(s_{i+1}, \mu'(s_{i+1} \mid \theta^{\mu'}) \mid \theta^{Q'})$;
12.	Update Critic's online network parameters θ^{Q} by formula (14);
13.	Update Actor's online network parameters θ^{μ} according to formula (15);
14.	Use formulas (16) and (17) to update Target network parameters;
15.	**End For**
16.	**End For**

4 Simulation Process and Result Analysis

(1) Convergence analysis

Fig. 4 shows how the system's utility changes with the iterations of training rounds. Each round needs to iterate 100 states for a total of 150 rounds of training. Because the RAPLSDDPG algorithm needs to store the data in the experience pool during the execution process, and then start to train the neural network after pre-stored part of the experience, the system utility rapidly rises from the 22nd round, and

gradually converges and tends to the 30th round. In a stable state, the system utility fluctuates slightly around -2.5, that is, the algorithm has a faster convergence speed, and can get a better solution for the goal of minimizing the average processing delay of the task.

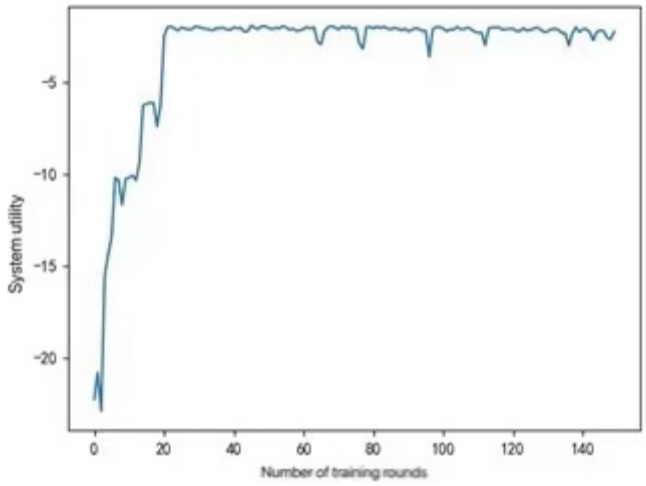

Fig. 4. Algorithm convergence curve

(2) Perform efficiency analysis

As shown in Fig. 5, the comparison results of the execution efficiency of the three algorithms are given. It can be seen from the figure that as the number of computing tasks increases, the execution time of the hierarchical joint optimal algorithm increases linearly. Although the optimal result can be obtained, the longer algorithm decision time will greatly reduce the user experience and weakened the advantages of task offloading technology in edge computing. Both the RAPLSDDPG algorithm and the multi-agent IDQL algorithm are based on deep reinforcement learning. After training, the model parameters can be loaded to quickly make task offloading and resource allocation decisions in the dynamically changing edge computing environment, which effectively reduces the complexity of the algorithm. Compared with the hierarchical algorithm, it can greatly reduce the execution time of the algorithm and improve the user experience.

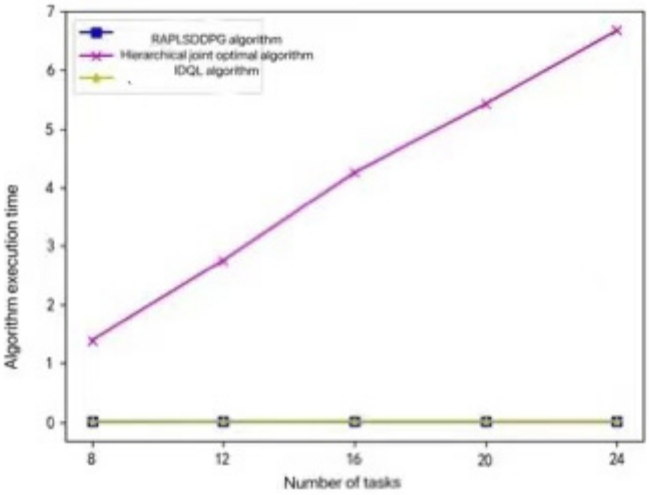

Fig. 5. Algorithm execution efficiency comparison

(3) Performance comparison and analysis

The relationship between system utility and safety outage probability is shown in Fig. 6. No matter which resource allocation algorithm is adopted, the system utility will increase with the increase of security outage probability. It can be seen from the graph that the system utility performance of RAPLSDDPG algorithm is similar to that of hierarchical joint optimization algorithm, the fluctuation is

relatively stable, and it can achieve near optimal results when the security outage probability is large. The decision-making scheme based on IDQL algorithm has less system utility, so the system utility is lower than the other two algorithms.

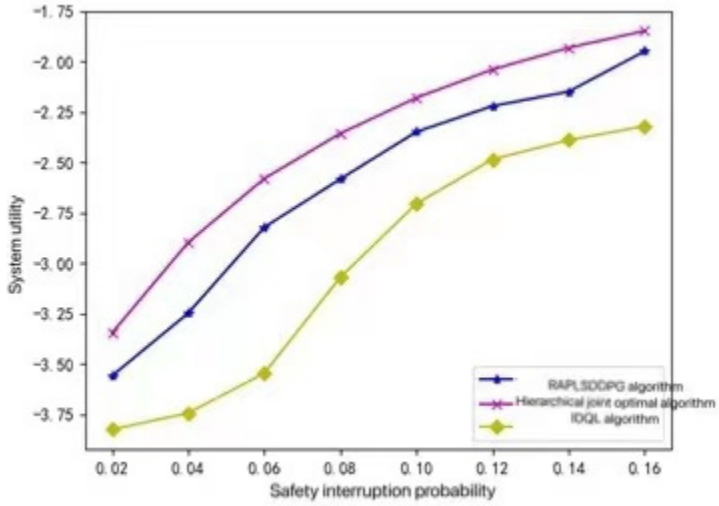

Fig. 6. The relationship between system utility and safety interruption probability

As shown in Fig. 7, the impact of the change in the security outage probability on the average unloading ratio is shown. Compared with the hierarchical joint optimal algorithm, the RAPLSDDPG algorithm can obtain relatively better results, and can obtain a near optimal unloading ratio when the security demand is small. The gap between the two algorithms is also within the affordable range of the user terminal. The IDQL algorithm has a limited number of actions to choose, so it cannot further optimize the system utility, so the unloading ratio is lower than the other two algorithms.

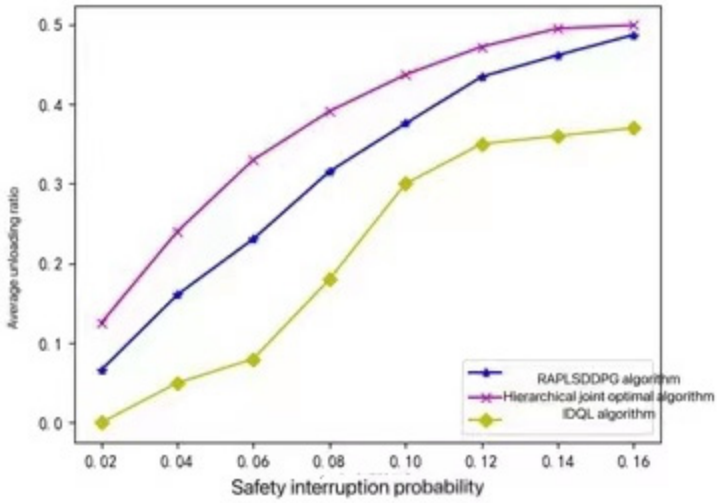

Fig. 7. The relationship between the average unloading rate and safety interruption probability

As shown in Fig. 8, when the resources of the edge server are limited, the performance of the RAPLSDDPG algorithm is almost the same as that of the layered algorithm. With the increase of the maximum amount of computing resources provided by the edge server, the system utility increases rapidly. At this time, the user terminal can choose a larger rate of task unloading to reduce the processing delay. When the computing resources of the edge server exceed 9 GHz, the system utility of the three algorithms almost tends to be stable. The system utility obtained by RAPLSDDPG algorithm and hierarchical joint optimal algorithm with the change of computing resources is similar, which is higher than that of IDQL algorithm.

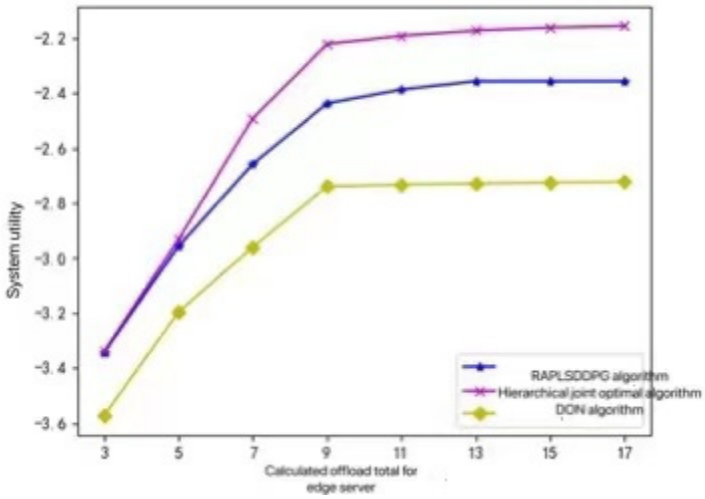

Fig. 8. The relationship between system utility and maximum computing resources

5 Conclusion

This article focuses on the modeling and analysis of resource allocation and security performance issues in edge computing scenarios. Aiming at the problem of privacy leakage caused by eavesdropping in the offloading process of computing tasks in single-cell and multi-user scenarios, physical layer security technology is adopted to develop confidentiality measures and establish a secure communication model. Considering that the wireless channel state of the system and the computing resources required by users are dynamically changing, the problem of partial migration of tasks of multiple user terminals is transformed into a joint optimization problem of computing resources and power resources under security settings and energy constraints. Aiming at the limitation that DQN algorithm can only deal with discrete actions and the low efficiency of hierarchical algorithm, the system is established as a Markov decision process, and a resource allocation algorithm RAPLSDDPG based on physical layer security and deep deterministic policy gradient is proposed. The paper uses the PyTorch framework to simulate the algorithm. The simulation results show that the RAPLSDDPG algorithm proposed in this paper can approach the optimal performance and has higher execution efficiency. It can quickly make a reasonable offloading strategy and resource allocation plan in a safe and dynamic environment. Effectively reduce the task processing delay, improve the user experience while ensuring the robustness of the system.

Acknowledgements

The research of the authors was supported by Industrial Internet innovation and development project of MIIT, China.

References

[1] C.-T. Ding, J.-N. Cao, L. Yang, S.-G. Wang, Overview of Edge Computing: Applications, Status Quo and Challenges, ZTE Technology Journal 25(03)(2019) 2-7.

[2] Tocze K., Nadjm-Tehrani S.A, Taxonomy for Management and Optimization of Multiple Resources in Edge Computing, Wireless Communications & Mobile Computing (2018)(2018) 1-23.

[3] W. Shi, H. Sun, J. Cao, Q. Zhang, W. Liu, A New Computing Model in the Internet of Everything Era, Computer Research and Development 54(05)(2017) 907-924.

[4] Satyanarayanan M., The Emergence of Edge Computing, Computer 50(1)(2017) 30-39.

[5] Y.-B. Mu, Y.-L. Chai, P. Song, L.-B. Bi, Wu Di, Research on the Development Status and Standard System of Edge Computing, Information and Communications Technologies 14(4)(2020) 23-30.

[6] C. Li, Research on Task Offloading Mechanism in Edge Computing, [dissertation] Beijing Jiaotong University, 2020.

[7] W. Shi, C. Jie, Z. Quan, Y. Li, L. Xu, Edge Computing: Vision and Challenges, Internet of Things Journal 3(5)(2016) 637-646.

[8] K. Peng, L. C. M. Victor, X.-L. Xu, L.-X. Zheng, J.-B. Wang, Q.-J. Huang, A Survey on Mobile Edge Computing: Focusing on Service Adoption and Provision, Wireless Communications and Mobile Computing (2018)(2018) 1-16.

[9] N Hassan, K. Yau, C. Wu, Edge Computing in 5G: A Review, IEEE Access (7)(2019) 127276-127289.

[10] J. Wang, L. Zhao, J. Liu, N. Kato, Smart Resource Allocation for Mobile Edge Computing: A Deep Reinforcement Learning Approach, IEEE Transactions on Emerging Topics in Computing 9(3)(2019) 1529-1541.

[11] X. Huang, Y. Rong, J. Kang, Z. Yan, Distributed Reputation Management for Secure and Efficient Vehicular Edge Computing and Networks, IEEE Access (5)(2017) 25408-25420.

[12] J.-B. Wang, H. Yang, M. Cheng, J.-Y. Wang, J.-Z. Wang, Joint Optimization of Offloading and Resources Allocation in Secure Mobile Edge Computing Systems, IEEE Transactions on Vehicular Technology 69(8)(2020) 8843-8854.

[13] H. Najmul, G. Saira, A. Ejaz, Y. Ibrar, I. Muhammad, The Role of Edge Computing in Internet of Things, IEEE Communications Magazine 56(11)(2018) 110-115.

[14] S. MUMTAZ, B. Ai, A. AL-DULAIMI, K.-F. TSANG, Guest Editorial 5G Tactile Internet: An Application for Industrial Automation, IEEE Transactions on Industrial Informatics 15(5)(2019) 2992-2994.

[15] X. Chen, D. W. K. Ng, W. H. Gerstacker, H.-H. Chen, A Survey on Multiple-Antenna Techniques for Physical Layer Security, IEEE Communications Surveys and Tutorials 19(2)(2017) 1027-1053.

[16] S. R. Aghdam, A. Nooraiepour, T. M. Duman, An Overview of Physical Layer Security With Finite-Alphabet Signaling, IEEE Communications Surveys and Tutorials 21(2)(2019) 1829-1850.