

## Research Article

# Blockchain-Enabled Joint Resource Allocation for Virtualized Video Service Functions

Ping Yu <sup>1,2</sup>, Hongwei Zhao <sup>1</sup>, Kun Yang,<sup>3</sup> Hanlin Chen,<sup>2</sup> Xiaozhong Geng,<sup>2</sup> Ming Hu,<sup>2</sup> and Hui Yan <sup>2</sup>

<sup>1</sup>College of Computer Science and Technology, Jilin University, Changchun 130000, Jilin, China

<sup>2</sup>Changchun Institute of Technology, Changchun 130012, Jilin, China

<sup>3</sup>Information and Communication Engineering, University of Electronic Science and Technology, Chengdu 611731, Sichuan, China

Correspondence should be addressed to Hui Yan; [yanhui7125@126.com](mailto:yanhui7125@126.com)

Received 7 May 2021; Revised 24 March 2022; Accepted 28 March 2022; Published 16 May 2022

Academic Editor: Yuanlong Cao

Copyright © 2022 Ping Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Power information is an important guarantee for energy security. As an important technical means of safety management and risk control, video monitoring is widely used in the power industry. Power video monitoring system uses efficient processing of multimodal video data and automatically identifies abnormal events and equipment status, replacing human monitoring with machine. Video monitoring data of power substations usually contain both visual information and auditory information, and the data types are diversified. The multimodal video data provides a rich underlying data source for the intelligent monitoring function, but it requires multiple service forms for efficient processing. Most intelligent edge monitoring equipment are only equipped with lightweight computing resources and limited battery supply, limited resources, and weak local processing data capabilities. Power video monitoring system has the characteristics of distribution, openness, interconnection, and intellectualization. Its intelligent edge video equipment is widely distributed, which also brings convenience and also brings security risks in terms of data security and reliability. For the outdoor multimodal power video monitoring system scenario, this paper adopts the edge-cloud distributed system architecture to solve the problem of resource shortage and adopts the first proposed service function virtualization (SFV) to solve the problem of multimodal video data processing. At the same time, the problem of security protection is solved by introducing blockchain to establish trust among intelligent video equipment and service providers. Under the security protection of virtualized service consortium blockchain (VSCB), virtualization technology is introduced into the service function chain (SFC) to realize SFV and solve the resource optimal allocation problem of multimodal video data processing. The work mainly involves the joint mapping of virtual resources, physical resources, and the joint optimization of computing and communication resources. Problems such as large state space and high dimensionality of action space have an impact on resource allocation. The stochastic optimization problem of resource allocation is established as a Markov decision process (MDP) model, and SFV technology is used to optimize cost and delay. The resource allocation optimization algorithm (RAOA-A3C) based on asynchronous advantage actor-critic algorithm (A3C) is proposed. Simulation experiments show that the RAOA-A3C proposed in this paper is more suitable for high-dynamic, multidimensional, and distributed power video monitoring system scenario and has achieved better optimization results in reducing time delay and deployment costs.

## 1. Introduction

Power is the energy basis for economic development, and power information is an effective guarantee for energy security. The power transformer is the most important core equipment in the operation of the power grid. If they fail, they

will have a significant impact. Therefore, real-time video monitoring and fault location of the transformer's operating states play a key role in ensuring the stable operation of the distribution network. For example, in order to ensure the long-term, high-efficiency, and safe operation of unattended or few-person-attended power equipment, the video

monitoring system have played an important role in the work of patrolling field equipment. As an important member of smart grid security protection, the power video monitoring system has the characteristics of distribution, openness, interconnection, and intelligence. First of all, most intelligent edge monitoring equipment equipped with limited battery supply and lightweight computing resources work without manual control. These equipment have obvious shortages of computing and storage resources and communication resources. Secondly, the interactive data of various types of intelligent edge monitoring equipment require multiple services to provide support. Thirdly, the widespread distribution of intelligent edge monitoring equipment makes security usually difficult to guarantee [1–4].

SFC is constituted by service requests for multimodal video data of intelligent edge monitoring equipment. SFV uses virtualization technology to realize the joint allocation of computing resources and communication resources.

The distributed power video monitoring system needs the support of edge computing technology. Edge computing is close to the edge of the network at the source of things or data and provides edge intelligent services nearby. It is an open platform that integrates core capabilities of network, computing, storage, and applications. It can meet the key demands of industry digitalization in agile connection, real-time business, data optimization, intelligence application, and so on [5–7]. Although the edge computing framework is suitable for application in the distributed power transformer video monitoring system scenario, it also faces challenges in terms of security.

The distributed power video monitoring system is a heterogeneous network, and its security needs the strong support of blockchain technology. Blockchain has the characteristics of smart contracts, distributed decision-making, collaborative autonomy, high security against tampering, openness, and transparency. Blockchain is similar to the power video monitoring system in terms of operation mode, topology, and especially security protection [8, 9]. Blockchain technology is an effective solution to establish trust among heterogeneous networks and realize reliable autonomous transaction management [10].

In view of the high dynamic and multidimensional characteristics of the power video monitoring system, Deep reinforcement learning (DRL) has gradually become a highly concerned optimization method [11–13]. DRL combines the perceptual ability of deep learning (DL) with the decision-making ability of reinforcement learning (RL). DRL is an artificial intelligence method that is closer to the way of human thinking and provides solutions for the perception and decision-making problems of complex systems [14]. DRL is suitable for solving complex optimization problems.

Edge computing systems can allocate resources to the edge of the network and provide low-delay network services for terminal equipment. However, there are still important issues such as resource management and safety protection in practical applications in the power video monitoring system scene [15].

It is very important to allocate resources reasonably and efficiently. Resources mainly include CPU resources, storage

resources, and communication resources. These resources are allocated by the controller to better solve problems such as cost and delay. In order to better improve the quality of service (QoS), edge computing and SFV are combined to decouple service functions from hardware equipment. SFV can realize flexible regulation and on-demand allocation of service resources [16, 17]. The use of mobile edge computing (MEC) technology can enable edge nodes to better achieve transaction autonomy [10]. Factors such as equipment heterogeneity, power supply status, and resource location of the power video monitoring system make the resource allocation more complicated. How to design the optimal resource allocation policy of SFC is a very challenging scientific issue [18]. DRL is widely regarded as an effective method to solve decision-making problems in complex environments [19, 20]. The SFC orchestration method based on DRL is used to solve the NP-hard problem of high-dimensional and intensive calculation [21–23]. DRL continuously interacts with the environment, automatically learns the optimal actions to be taken in different states, and optimizes resource allocation according to the optimal strategy.

The heterogeneity of edge nodes makes edge computing more complex and uncertain [24, 25]. The central control node may also suffer a single point of failure, which may cause data to leakage or malicious tampering and ultimately lead to task execution failure or economic loss [26]. Both the equipment themselves and the communications among equipment are facing threats of various security attacks. For example, the equipment may malfunction or be malicious so that the transmitted information may be leaked or tampered with. Therefore, it is very important to ensure data security. Blockchain is a kind of cryptology-supported, verifiable, and immutable ledger. Blockchain ensures interaction through transaction records and distributed consensus on the validity of transaction records. Blockchain with the characteristics of pan-central, distributed, and trustworthy provides new ideas for designing the framework and paradigm of cloud-edge computing [27].

In summary, the integration of SFV, blockchain, edge computing, and DRL technology to solve the resource allocation optimization problem of the power video monitoring system is very worthy of discussion.

## 2. Related Work

At present, there have been some papers that combine edge computing with blockchain. The introduction of blockchain can solve the security problem of the cloud-edge computing environment. Reference [28] proposes to use blockchain for decentralized task allocation and scheduling in MEC. The purpose is to eliminate the increase in the computational burden of the central server due to the attacker's distributed denial of service attack, so it affects the accuracy of data transmission. Reference [29] proposes an internet of vehicles (IoV) file-sharing scheme based on blockchain smart contracts and attribute encryption. Under the premise of ensuring the efficiency of filesharing, the file-sharing solution

adopts blockchain smart contract technology to avoid third-party participation and protect data security. Reference [30] proposes a blockchain-based energy transaction framework for energy transactions among electric vehicles and smart grids. The autonomous and controllable consensus mechanism puzzle generated by the edge server helps increase transaction speed. References [31, 32] propose that edge servers and terminal equipment participate in the blockchain, and the consortium blockchain is used to manage the virtual resources. Users registered in the consortium blockchain can define and deploy their own virtual systems and read and write blocks. The allocation mechanism of the virtual network function (VNF) improves the efficiency of resource allocation while ensuring security. Reference [33] uses a static VNF allocation policy to reduce the cost of operators while ensuring users' QoS. However, the network environment is dynamically changing, and it is more reasonable to consider long-term optimization. Reference [34] achieves the purpose of reducing end-to-end delay by reducing transmission delay and processing delay, but it does not pay attention to the utilization of physical network resources. Reference [35] realizes the reduction of service provider's capital expenditure and operating expenditure, but it sacrifices reliability and does not consider the end-to-end delay. Reference [36] proposes an algorithm based on deep Q-learning (DQL) to solve the decision-making problem of computing resource allocation at the edge of a multiuser shared network. Reference [37] applies the DRL algorithm to jointly optimize the computational efficiency of the MEC system and the transaction throughput of the blockchain system for the industrial internet system based on the blockchain.

Although the above papers have optimized the security and system performance of the cloud-edge computing environment to varying degrees, there are few related studies in the power video monitoring system scenario, and there are still some potential problems and challenges. First of all, although the introduction of blockchain technology can solve the security problem, the consensus process in the blockchain is inefficient, and there is a serious computational overhead in the system. Secondly, resource allocation still has the following problems. For example, most studies in many papers are based on the prerequisite of the known state of the environment and do not take into account the dynamic changes in the environment over time. Nor does it take into account the fact that the arrival of a large number of service requests will easily cause a backlog of service requests, which will affect the stability of the network. It also failed to take into account the user's QoS while optimizing the cost of resource allocation. Thirdly, there are also problems in solving optimization. The continuous increase in the number of agents will explode the dimensions of the state space, and it becomes infeasible to use the traditional tabular method to solve the problem. DRL can solve the problem of state space explosion caused by the increase in the number of nodes [38, 39]. DRL has been proven to effectively approximate the Q value of RL by using a deep neural network (DNN) [20]. The goal of this paper is to

achieve low-delay, low-cost resource optimization through the use of blockchain, DRL, and SFV technology in the cloud-edge computing environment.

A power video monitoring system is a distributed heterogeneous cloud-edge network. The key to solving the problem is how to select server and physical links that meet service requirements from limited physical resources for allocation [40, 41]. The goal is to maximize resource utilization while ensuring network performance. This paper combines edge computing and SFV to build a cloud-edge computing basic model in order to achieve transaction autonomy at the edge and achieve better QoS. Power video monitoring system is a distributed heterogeneous network involving different public and private networks. The unreliability is obvious. This paper introduces blockchain technology to achieve reliable transaction autonomy. The resource optimization allocation problem is an intensive calculation problem. It is a high-dimensional NP-hard problem. This paper introduces DRL technology to solve the NP-hard problem. In summary, the resource allocation optimization problem is modeled as an MDP, and the resource allocation policy is optimized through SFV to maximize long-term utility performance. This paper proposed the RAOA-A3C algorithm based on A3C in order to obtain the optimal resource allocation policy and finally achieved the goal of improving safety protection and efficient resource management.

The main contributions of this paper are as follows:

- (1) The SFV concept was first proposed based on the characteristics of the power video monitoring system. Multimodal video data service requests constitute service function chains, which use SFV technology to optimize the allocation of computing resources and communication resources.
- (2) SFV, blockchain, edge computing, and DRL technology are used to solve the resource allocation optimization problem of the power video monitoring system. The optimization problem mainly involves the joint mapping of virtual resources and physical resources and the joint optimized allocation of computing resources and communication resources.
- (3) The system architecture is built. The proposed VSCB solves the problem of safety protection. The random optimization problem of resource allocation is modeled as an MDP model, and the RAOA-A3C algorithm is proposed. Simulation experiments show that the delay and cost of the RAOA-A3C algorithm are superior to other methods.

The structure of this paper is arranged as follows. Section 1 introduces the background. Section 2 introduces related work. Section 3 gives the system architecture and workflow. Section 4 proposes the system model. Section 5 proposes the optimization algorithm. Section 6 introduces performance evaluation and analysis. Section 7 summarizes the work.

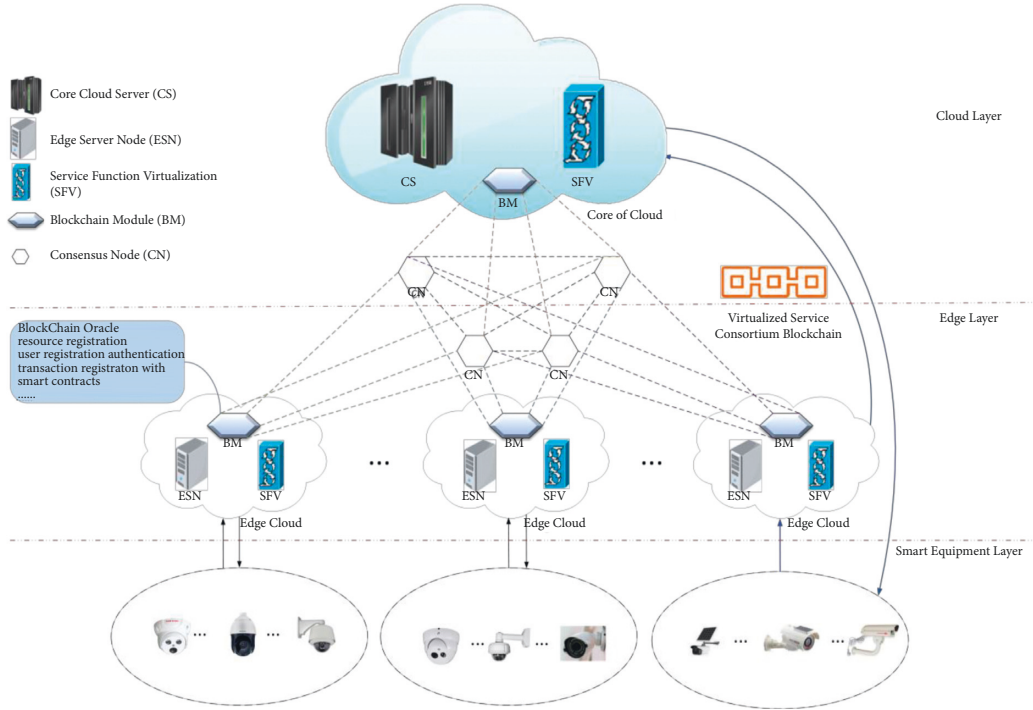


FIGURE 1: Blockchain-enabled system architecture.

### 3. Blockchain-Enabled System Architecture and Work Flow

3.1. *Blockchain-Enabled System Architecture.* The system architecture of the power video monitoring system is shown in Figure 1.

This paper combines the requirements of the power video monitoring system in resource management and safety protection to build a system architecture. This architecture mainly includes the following three layers:

- (1) **Intelligent Equipment Layer:** The intelligent equipment layer containing multiple types of equipment is at the bottom. Intelligent equipment mainly complete the work of data acquisition and data intelligent processing. Because the resources of the intelligent equipment layer are limited, the intelligent equipment layer that filters out data that have been processed locally send service requests to edge clouds or the core cloud layer.
- (2) **Edge Layer:** The edge layer is composed of heterogeneous edge clouds. The distribution and heterogeneity of the edge layer make the traditional edge layer unable to guarantee the reliability of the service. The edge layer applied with VSCB has the ability to ensure service consistency and provide reliable service management. Each edge cloud in the system model contains three components: (1) service node, (2) blockchain module, and (3) controller.
- (3) **Cloud Layer:** The core cloud layer and the edge layer reach consensus in the same blockchain. The power

video monitoring system belongs to a distributed heterogeneous cloud-side computing environment involving public and private networks. Its system architecture uses VSCB to build a trusted cloud-side computing environment. When the resources of the edge layer cannot meet the service quality and resource constraints of the terminal equipment, it can continue to send service request information to the core cloud in order to obtain relevant resources of the core cloud platform to complete the current service request. The cloud layer also mainly includes three components: (1) service node, (2) blockchain module, and (3) controller.

Next, the three main components included in both the cloud layer and the edge layer are introduced.

- (1) **Service Nodes:** Service nodes of the edge cloud and the core cloud are mainly composed of servers. Each server node is the actual host of the virtual service functions (VSF), which specifically provides various resource services.
- (2) **Blockchain Module:** The blockchain module is composed of high-performance equipment or other lightweight equipment. It is responsible for resource registration, user registration, authentication, smart contract, and transaction registration to ensure trusted and reliable resource allocation [10].
- (3) **Controller:** The controller mainly includes SFV. The essence of SFV is to turn dedicated hardware equipment into general software equipment to achieve the purpose of sharing hardware



infrastructures. The software equipment called VSFs realize functions such as the rapid establishment of the network among VSFs and the rapid allocation of resources. The quality of resource allocation by the controller affects the efficiency of service provision and physical resource usage [42]. The resource allocation optimization algorithm is deployed in the controller. The controller manages, allocates, and monitors the underlying resources. The controller obtains the system information reported from the bottom layer; analyzes the network topology, equipment operation energy consumption, resource utilization, and so on; and then performs tasks such as resource mapping, traffic scheduling, and policy management [21]. The controller helps improve the efficiency of resource management.

**3.2. Consortium Blockchain.** Blockchain is a kind of chained data structure that combines data blocks sequentially in a time sequence. It is a distributed ledger that cannot be tampered with and cannot be forged, and it is guaranteed by cryptography [43, 44]. The data of the blockchain is collectively maintained. Data operations are witnessed and stored by all nodes, so the data cannot be changed, and it is safe and reliable [45, 46]. Blockchain is a technology that realizes information security and information transparency based on a consensus mechanism. The consortium blockchain is a relatively new way of applying blockchain technology to businesses. It is suitable for providing services for joint collaboration among multiple enterprises, and it has the characteristics of partial decentralization. The consensus participants of the consortium blockchain are a group of preapproved nodes on the network, and the consortium blockchain can exercise a greater degree of control over the network.

Blockchain has been widely used in many fields. Resource allocation in the cloud-edge computing environment is one of the typical cases. The VSCB system proposed in this paper is based on a limited number of enterprises to form the consortium blockchain, and the number of nodes is also limited. Even if there is an expansion of nodes, it will not increase infinitely. The workflow trusted authentication mechanism of VSCB proposed in this paper is described as follows: For any node in the system, its operation is limited by the role control and permission control information on the consortium blockchain to limit its operation scope. The node can read the role control and permission control information to ensure that its work is legal. When the node completes the work and writes the flow information, the role control and permission control are authenticated on the entire consortium blockchain to ensure the normal operation of the entire workflow. At the same time, when the node wants to operate, it must reach a consensus on its authority on the consortium blockchain before writing its operation into the consortium blockchain. When the workflow

continues to flow to the next link, if there is a problem with the authority, then data writing and workflow flow cannot proceed normally.

This paper adopts the practical Byzantine fault tolerance (PBFT) algorithm. The advantages are: first, the system can be separated from the existence of encrypted tokens, the nodes of the algorithm consensus are composed of business participants or supervisors, and the security and stability are guaranteed by business stakeholders. Secondly, the time delay of consensus is short, which basically meets the requirements of commercial real-time processing. Thirdly, the consensus efficiency is high, which can meet the needs of the high-frequency trading volume. Moreover, because of the independence of the smart contract, its execution process and the generated transaction information will not be "maliciously polluted" by the outside world on the consortium blockchain, making the credibility of the transaction information far more than that of the public blockchain. Therefore, the consortium blockchain adopts a more competitive PBFT algorithm, which can improve the application level of the consortium blockchain at the enterprise level to a new level.

This paper uses the token-free optimized PBFT algorithm [10]. The master node is not determined by complex computing puzzles, the master node is determined by circular selection. Therefore, this optimization algorithm can better meet the needs of the power video monitoring system in terms of saving resources.

**3.3. Work Flow.** The workflow is roughly described as follows: In the cloud-edge computing environment, resources are registered as digital assets on the VSCB, and resource management is realized through the controller. The two main events of this system architecture are resource registration and resource allocation.

**3.3.1. Resource Registration.** The core cloud or edge cloud needs to register resource information on the blockchain module before providing services. They send information such as equipment identification and related attributes to the blockchain module. The blockchain module is maintained a list of information to form a resource pool and then uses this information to form a block. In this way, the core cloud or edge cloud can provide hosts for VSFs under the supervision of the VSCB.

**3.3.2. Resource Allocation.** When the smart equipment layer sends out a service request, the current request is first allocated to the adjacent edge cloud in the edge layer. The blockchain module in the edge cloud first verifies the user's identity. After the identity of the user who sent the service request is authenticated, the request is passed to the controller to obtain the optimal resource allocation. If the resource constraints of the adjacent edge cloud and the QoS of the user cannot be guaranteed, the service request is sent to

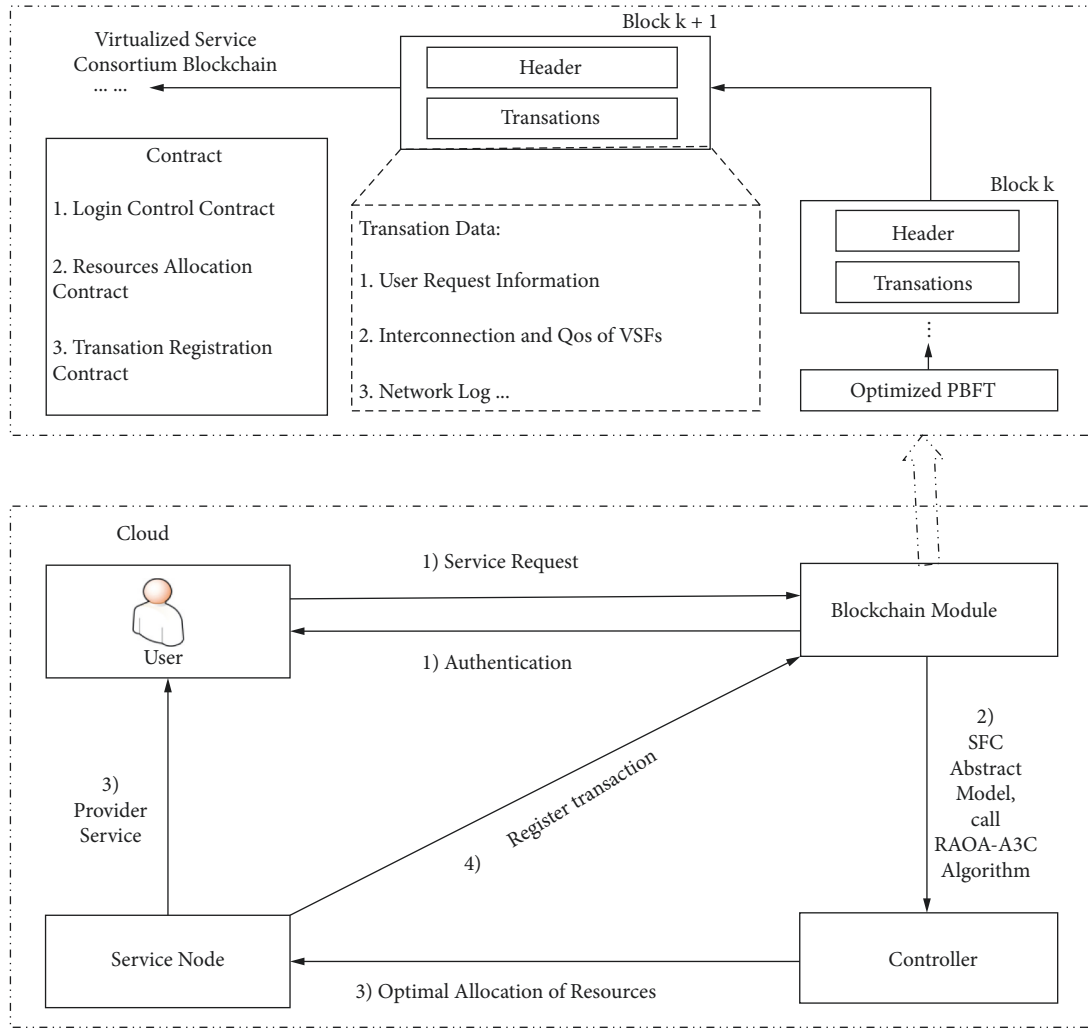


FIGURE 2: Resource allocation workflow.

the controller on the core cloud layer to obtain the optimal resource configuration.

Since the important component functions of the core cloud are the same as those of the edge cloud, their workflow is shown in Figure 2. The process includes the following four steps:

- (1) **User Registration and Authentication:** User information related to equipment identification, encryption data keys, and equipment attributes needs to be registered on the blockchain module. After a user sends a service request, the user's information will be authenticated.
- (2) **Resource Optimization Department:** The intelligent equipment sends a service request to the controller, and the service request invokes the RAOA-A3C algorithm in the controller to obtain the optimal resource allocation.

- (3) **Provide Services:** The controller controls the relevant service nodes to provide services to users according to the optimal resource allocation.
- (4) **Transaction Registration:** The registration transactions that include information such as interconnection, attributes, sequence of virtual files, user information, and QoS trigger the smart contract on the VSCB. The registration transaction executes the consistent process of the optimized PBFT algorithm. A new block is generated, the resource allocation transaction takes effect, and the trusted service is completed.

This paper uses the RAOA-A3C algorithm to achieve the parallel execution effect. Each controller as an agent extracts state information from the environment, and then the controller obtains the action probability by processing the state information, and then the controller calculates the

reward based on the agent's action. During the interaction between the controller and the local environment, the agent updates its local action probability according to the reward and regularly pushes its gradient to the global network.

#### 4. System Model and Problem Formulation

The physical network of the optimized model in this paper mainly involves servers and physical links. They provide instantiated computing resources including CPU resources and storage resources and instantiated bandwidth resources for the VSFs that constitute each SFC. In this paper, CPU resources are used to represent computing resources. The physical network feeds back the current CPU resources and bandwidth resources to the RAOA-A3C in the controller. The algorithm makes decisions based on the current CPU resource status of the node, the current resource status of the link, and the current queue status in the SFC. Then the controller optimizes the resource allocation policy through the resource management entity [38]. This section introduces the network model, service request model, cost model, delay model, and optimization goals.

**4.1. Network Model.** The physical network is abstracted as an undirected graph  $UG^P = (SN^P, Y^P)$ , where  $SN^P$  represents a collection of nodes, the nodes are divided into two categories: (1) server nodes  $n \in N^P$ , which provide instantiated CPU resources for VSFs, and each server can instantiate multiple VSFs. And (2) switch nodes, which forward the traffic.  $Y^P$  represents a collection of physical links.  $C_n^P$  represents the CPU capacity of each server  $n$ . In order to ensure the resource utilization of the server and achieve the purpose of energy-saving, it is necessary to set a CPU resource threshold  $\varphi_n^P$  for the server. As long as the remaining CPU resources of the server in each time slot are less than  $\varphi_n^P$ , the server can be used.  $B_{mn}^P$  represents the bandwidth capacity of the physical link  $mn$  connecting adjacent servers  $n$  and  $m$ .

**4.2. Service Request Model.**  $SF$  represents the collection of SFCs. The  $i$ -th SFC can be formalized as an undirected graph  $UG_i^N = (V_i^N, L_i^N)$ , where  $V_i^N$  represents the collection of different types of VSFs on the  $i$ -th SFC and  $L_i^N$  represents the collection of virtual links on the  $i$ -th SFC.  $V_{i,j}^N$  represents the  $j$ -th VSF on the  $i$ -th SFC;  $C_{i,j}^N$  represents the CPU resource allocated by the server to the  $j$ -th VSF on the  $i$ -th SFC; and  $B_{i,jk}^{mn}$  represents the virtual link bandwidth resource allocated by the physical link to the adjacent VSF  $jk$  on the  $i$ -th SFC.  $D_i$  represents the maximum delay limit of the  $i$ -th SFC.  $\theta_{i,j}^n$  represents the mapping of VSF to the server, which is a Boolean variable.  $\theta_{i,j}^n = 1$  represents the  $j$ -th VSF on the  $i$ -th SFC mapping the server  $n$ , and  $\theta_{i,j}^n = 0$  represents no mapping relationship.  $\eta_{i,jk}^{mn}$  represents the mapping of virtual links to physical links, which is also a Boolean variable.  $\eta_{i,jk}^{mn} = 1$  represents the virtual link  $l_{i,jk}^{jk}$  connecting the adjacent VSF  $jk$  on the  $i$ -th SFC that is mapped to the physical link  $y^{mn}$  connecting the adjacent server  $mn$ , and  $\eta_{i,jk}^{mn} = 0$

represents no mapping relationship. This paper makes the following constraints.

In time slot  $t$ , each VSF  $V_{i,j}^N$  can only select one server for mapping. That is,

$$\sum_{n \in N^P} \theta_{i,j}^n(t) = 1, \quad \forall i \in SF, \forall j \in V_i^N. \quad (1)$$

The binary variable that represents the mapping of VSF to the server is expressed as follows:

$$\theta_{i,j}^n(t) \in \{0, 1\}, \quad \forall i \in SF, \forall j \in V_i^N, \forall n \in N^P. \quad (2)$$

In time slot  $t$ , the amount of CPU resources allocated by the server should not exceed its CPU capacity  $C_n^P$  so that the system stability can be guaranteed. That is,

$$\sum_{i \in SF} \sum_{j \in V_{i,j}^N} \theta_{i,j}^n(t) C_{i,j}^n(t) \leq C_n^P, \quad \forall n \in N^P. \quad (3)$$

In time slot  $t$ , the remaining CPU capacity  $R_n(t)$  of the server  $n$  can be expressed as the CPU capacity  $C_n^P$  minus the amount of CPU resources. That is,

$$R_n(t) = C_n^P - \sum_{i \in SF} \sum_{j \in V_{i,j}^N} \theta_{i,j}^n(t) C_{i,j}^n(t), \quad \forall n \in N^P. \quad (4)$$

And the constraint is as follows:

$$R_n(t) \leq \varphi_n^P, \quad \forall n \in N^P. \quad (5)$$

In time slot  $t$ , each virtual link  $l_{i,jk}^{jk}$  connected to adjacent VSF  $jk$  can only select one physical link  $y^{mn}$  connected to adjacent server  $mn$  for mapping. That is,

$$\sum_{n,m \in N^P} \eta_{i,jk}^{mn}(t) = 1, \quad \forall i \in SF, \forall j, k \in V_i^N. \quad (6)$$

The binary variable that represents the mapping of virtual links to physical links is expressed as follows:

$$\eta_{i,jk}^{mn}(t) \in \{0, 1\}, \quad \forall i \in SF, \forall j, k \in V_i^N, \forall n, m \in N^P. \quad (7)$$

In time slot  $t$ , the amount of bandwidth resources allocated by physical link  $mn$  cannot exceed its bandwidth capacity  $B_{mn}^P$ . That is,

$$\sum_{i \in SF} \sum_{j,k \in V_{i,jk}^N} \theta_{i,j}^n(t) \theta_{i,k}^m(t) \eta_{i,jk}^{mn}(t) B_{i,jk}^{mn}(t) \leq B_{mn}^P, \quad \forall n, m \in N^P. \quad (8)$$

In time slot  $t$ , the remaining bandwidth resource  $R_{mn}(t)$  can be expressed by the bandwidth capacity  $B_{mn}^P$  minus the bandwidth resource. That is,

$$R_{mn}(t) = B_{mn}^P - \sum_{i \in SF} \sum_{j,k \in V_{i,jk}^N} \theta_{i,j}^n(t) \theta_{i,k}^m(t) \eta_{i,jk}^{mn}(t) B_{i,jk}^{mn}(t), \quad \forall n, m \in N^P. \quad (9)$$

**4.3. Cost Model.** The allocation cost of resource allocation mainly includes the cost of occupying CPU resources  $OC_{i,j}^n$  and the cost of occupying physical link bandwidth resources  $OB_{i,jk}^{mn}$  [47].  $OC_{i,j}^n(t)$  is inversely proportional to the

remaining CPU resource  $R_n(t)$  of server  $n$  in time slot  $t$ . That is,

$$OC_{i,j}^n(t) = \frac{\alpha}{R_n(t)}, \quad (10)$$

where  $\alpha$  is a positive number.

$OB_{i,jk}^{nm}$  is inversely proportional to the remaining bandwidth resources  $R_{nm}(t)$  of physical link  $nm$ , that is,

$$OB_{i,jk}^{nm}(t) = \frac{\beta}{R_{nm}(t)}, \quad (11)$$

where  $\beta$  is a positive number.

In summary, in time slot  $t$ , the resource allocation cost on the  $i$ -th SFC is

$$\begin{aligned} O_i(t) &= \sum_{j \in V_i^N} \sum_{n \in N^P} \theta_{i,j}^n(t) \frac{\alpha}{R_n(t)} \\ &+ \sum_{j,k \in V_i^N} \sum_{n,m \in N^P} \theta_{i,j}^n(t) \theta_{i,k}^m(t) n_{i,jk}^{mn}(t) \frac{\beta}{R_{nm}(t)}, \quad \forall i \in SF. \end{aligned} \quad (12)$$

**4.4. Delay Model.** The optimization model not only gives the attributes and order of VSFs but also provides QoS constraints. The delay of this model mainly considers the queuing delay, processing delay, and link transmission delay. We take the  $i$ -th SFC as an example;  $Q_i(t)$  represents its queue length in time slot  $t$ ;  $P_i(t)$  represents the size of the data packet, and it is assumed that the size of the data packet obeys the exponential distribution of parameter  $\bar{P}$ ; and  $A_i(t)$  represents the data packet arrival process of the  $i$ -th SFC, and it is assumed that the arrival of data packets obeys the Poisson distribution with a parameter of  $\lambda_i$  [48]. The update process of the queue is expressed as follows:

$$Q_i(t+1) = \max[Q_i(t) - \ell_{i,1}(t) + A_i(t), 0], \quad (13)$$

where  $\ell_{i,1}(t)$  represents the first VSF service rate of the  $i$ -th SFC and the service rate  $\ell_{i,j}(t)$  of the  $j$ -th VSF on the  $i$ -th SFC is determined by the amount of CPU resources allocated to the  $j$ -th VSF by the server, that is,  $\ell_{i,j}(t) = C_{i,j}^n \cdot \varepsilon$ , where  $\varepsilon$  is the service rate coefficient, which represents the ratio between CPU resources and service rate [49]. The constraints of the delay model are as follows.

$Q_i^{\max}(t)$  represents the maximum queue length of the  $i$ -th SFC. In order to ensure that the queue length does not overflow,  $Q_i(t)$  satisfies

$$Q_i(t) < Q_i^{\max}(t), \quad \forall i \in SF. \quad (14)$$

$T_i^{\text{wait}}$  represents the queuing delay of the  $i$ -th SFC. According to little theorem, the queuing delay  $T_i^{\text{wait}}$  is

$$T_i^{\text{wait}}(t) = \frac{Q_i(t)}{\lambda_i(t)}. \quad (15)$$

$T_{i,j}^{\text{proc}}$  represents the processing delay generated by each VSF, and  $T_{i,j}^{\text{proc}}$  is

$$T_{i,j}^{\text{proc}}(t) = \frac{DR_{i,j}(t)}{\ell_{i,j}(t)}. \quad (16)$$

$DR_{i,j}(t)$  represents the amount of data packets arriving at VSF  $V_{i,j}^N$  in time slot  $t$ , and the processing delay  $T_i^{\text{proc}}(t)$  of the  $i$ -th SFC is

$$T_i^{\text{proc}}(t) = \sum_{j \in V_i^N} T_{i,j}^{\text{proc}}(t). \quad (17)$$

$T_{i,jk}^{\text{tran}}$  represents the transmission delay of the amount of data.  $T_{i,jk}^{\text{tran}}$  is related to the amount of data transmitted and the bandwidth resources allocated by the physical link.  $T_{i,jk}^{\text{tran}}$  is

$$T_{i,jk}^{\text{tran}}(t) = \frac{DR_{i,jk}(t)}{B_{i,jk}^{nm}(t)}. \quad (18)$$

$DR_{i,jk}(t)$  represents the amount of data from VSF  $V_{i,j}^N$  to VSF  $V_{i,k}^N$ , that is, the amount of data packets arriving at VSF  $V_{i,k}^N$  in time slot  $t$ . The transmission delay  $T_i^{\text{tran}}(t)$  of the  $i$ -th SFC in time slot  $t$  is

$$T_i^{\text{tran}}(t) = \sum_{j,k \in V_i^N} T_{i,jk}^{\text{tran}}(t). \quad (19)$$

In summary, the total delay  $T_i(t)$  of the  $i$ -th SFC is

$$T_i(t) = T_i^{\text{wait}}(t) + T_i^{\text{proc}}(t) + T_i^{\text{tran}}(t). \quad (20)$$

And the constraints are as follows:

$$T_i(t) < D_i, \quad \forall i \in SF. \quad (21)$$

**4.5. Optimization Goals.** The main optimization goal of this paper is to minimize the cost of resource allocation under the premise of ensuring security and meeting the requirements of delay. CPU resources and physical link bandwidth resources are reasonably allocated, which is conducive to the realization of low-delay and low-cost resource allocation. The utility function  $U(t)$  is defined as follows:

$$U(t) = -e1 \frac{\sum_{i \in SF} O_i(t)}{O_{\max}} - e2 \frac{\sum_{i \in SF} T_i(t)}{\sum_{i \in SF} D_i(t)}, \quad (22)$$

where  $e1$  and  $e2$  are the weight values and  $e1 + e2 = 1$ .  $O_{\max}$  represents the maximum value of the allocation cost. After the algorithm normalizes the allocation cost, the optimization goal is expressed as follows:



$$\begin{aligned}
& \max_{\theta_{i,j}^n(t), \theta_{i,k}^m(t), \eta_{i,jk}^{mn}(t), C_{i,j}^n(t), B_{i,jk}^{mn}(t)} U(t), \\
& \text{s.t. C1: } \sum_{n \in N^P} \theta_{i,j}^n(t) = 1, \quad \forall i \in SF, \forall j \in V_i^P, \\
& \text{C2: } \sum_{m,n \in N^P} \eta_{i,jk}^{mn}(t) = 1, \quad \forall i \in SF, \forall j, k \in V_i^P, \\
& \text{C3: } \sum_{i \in SF} \sum_{j \in V_{i,j}^N} \theta_{i,j}^n(t) C_{i,j}^n(t) \leq C_n^P, \quad \forall n \in N^P, \\
& \text{C4: } \sum_{i \in SF} \sum_{j,k \in V_{i,j}^N} \theta_{i,j}^n(t) \theta_{i,k}^m(t) \eta_{i,jk}^{mn}(t) B_{i,jk}^{mn}(t) \leq B_{nm}^P, \quad \forall n, m \in N^P, \\
& \text{C5: } R_n(t) \leq \phi_n^P, \quad \forall n \in N^P, \\
& \text{C6: } Q_i(t) < Q_i^{\max}(t), \quad \forall i \in SF, \\
& \text{C7: } T_i(t) < D_i, \quad \forall i \in SF, \\
& \text{C8: } \theta_{i,j}^n(t) = \{0, 1\}, \quad \forall i \in SF, \forall j \in V_i^N, \forall n \in N^P, \\
& \text{C9: } \eta_{i,jk}^{mn}(t) = \{0, 1\}, \quad \forall i \in SF, \forall j, k \in V_i^N, \forall n, m \in N^P,
\end{aligned} \tag{23}$$

where C1 guarantees that each VSF in the virtual network can only select one server in the physical network for mapping. C2 guarantees that the virtual link of adjacent VSFs can only select the physical link of adjacent servers in the physical network for mapping. C3 guarantees that the sum of the CPU resources allocated by each server cannot exceed the CPU capacity of the server. C4 makes the sum of all communication resources mapped to a certain physical link not exceed the total bandwidth resources of the physical link. C5 makes the remaining CPU resources of each server lower than the threshold, guarantees the resource utilization of the server, and further achieves the effect of energy-saving. C6 guarantees that the queue length of each SFC does not overflow. C7 guarantees that each SFC must meet the delay requirement in any time slot. C8 and C9 are requirements for binary variables. In summary, the utility function is restricted by the C1–C7 constraints to ensure the effectiveness of the optimization objective.

## 5. Proposed Algorithm

In this section, the resource allocation optimization problem of the power video monitoring system is modeled as an MDP model, and then the RAOA-A3C algorithm is proposed in the cloud-edge computing environment to achieve the goals of security protection and efficient resource management.

**5.1. Problem Transformation.** The MDP model mainly includes state space, action space, transition probability, and reward function [38].

**5.1.1. State Space.**  $S$  represents the state space, which is mainly composed of the queue status of each SFC, the remaining CPU resources of the server, and the remaining resources of the physical link bandwidth.  $s(t)$  represents the

state of the network in time slot  $t$ , which is expressed as follows:

$$s(t) = \{Q_i(t), R_n(t), R_{nm}(t)\} \quad \forall m, n \in N^P, \forall i \in SF. \tag{24}$$

**5.1.2. Action Space.**  $A$  represents the action space, which mainly includes allocating CPU resource  $C_{i,j}^n(t)$ , allocating bandwidth resource allocation  $B_{i,jk}^{mn}(t)$ , and deploying  $\theta_{i,j}^n(t)$  and  $\eta_{i,jk}^{mn}(t)$ .  $a(t)$  represents the action taken by the network in time slot  $t$ , which is expressed as follows:

$$a(t) = \{C_{i,j}^n(t), B_{i,jk}^{mn}(t), \theta_{i,j}^n(t), \eta_{i,jk}^{mn}(t)\} \quad \forall n, m \in N^P, \tag{25}$$

$$\forall i \in SF, \forall j, k \in V_i^N.$$

**5.1.3. Transition Probability.** In time slot  $t$ , there is a probability that the network state  $s(t)$  takes action  $a(t)$  and transitions to the network state  $s(t+1)$ .  $\Pr(s(t), a(t), s(t+1))$  represents the transition probability, and  $\Pr(s(t), a(t), s(t+1))$  is expressed as follows:

$$\begin{aligned}
& \Pr(Q_i(t), a(t), Q_i(t+1)) \\
& \Pr(s(t), a(t), s(t+1)) = \Pr(R_n(t), a(t), R_n(t+1)) \cdot \\
& \Pr(R_{nm}(t), a(t), R_{nm}(t+1))
\end{aligned} \tag{26}$$

**5.1.4. Reward Function.** This section uses the aforementioned utility function as a reward.  $r(s(t), a(t))$  represents the reward function, which is the reward after the network state  $s(t)$  takes an action  $a(t)$ .  $r(s(t), a(t))$  can be expressed as follows:

$$r(s(t), a(t)) = U(t). \tag{27}$$

**5.2. Algorithm Description.** Based on formula (23) and the MDP model, the key problem to be solved in this paper is to determine the target server and resource allocation policy. The algorithm obtains the optimal value function of the state of each slot and then obtains the optimal action corresponding to the state, that is, the optimal action of each slot constitutes the optimal policy  $\pi^*$ . This paper introduces the DRL algorithm to solve the optimization problem of the MDP model. DRL uses DNN to effectively identify high-dimensional state spaces and uses RL algorithms to learn complex tasks in an end-to-end manner. DRL does not require complicated manual preprocessing of state features.

Most intelligent monitoring equipment have the capability of parallel computing. A3C is an asynchronous actor-critic parallel learning algorithm based on the advantage function. It is a lightweight DRL framework. The framework uses an asynchronous gradient descent method to optimize the parameters of the controller, which is suitable for solving the problems of too large state space and the high dimension of action space in the optimal allocation of resources. This paper proposes the RAOA-A3C algorithm to solve the MDP model.

As shown in Figure 2, after the user sends a service request and performs security authentication, the controller in the server node collects the environment status and takes actions to react to the status. The general workflow of the algorithm is shown in Figure 3. The environment state is provided to the actor network and the critic network, and the policy and the value function are obtained, respectively. The actor executes the action, and then the critic evaluates whether the action is good or bad. The policy  $\pi$  is a function of state  $s$ , which returns the probability distribution of all actions, and sums up to 1. That is,  $\pi(a|s)$  represents the probability of choosing action  $a(t)$  in state  $s(t)$ . In the actual execution process, the actor selects actions based on the distribution of policy  $\pi(a(t)|s(t); \theta)$  or directly selects the action with the highest probability. Accordingly, the critic evaluates the current policy based on the TD error between the value function  $V(s(t); \theta_v)$  and the current reward, where  $\theta$  is the actor network parameter,  $\theta_v$  is the critic network parameter, and TD-error is used to update  $\theta$  to correct the action probability;  $\theta_v$  can improve the accuracy of the value evaluation.

The algorithm uses the following iterative definition as the value function  $V(s)$  of the expected discount return:

$$V(s) = E_{\pi(s)}(r + \gamma V(s')). \quad (28)$$

The return obtained in the current state is the sum of the return obtained in the next state and reward  $r$  obtained during the state transition, where  $\gamma$  represents the discount factor in RAOA-A3C.

There is also an action value function  $Q(s, a)$  closely related to the value function, which is defined as follows:

$$Q(s, a) = r + \gamma V(s'). \quad (29)$$

The advantage function  $A(s, a)$  is defined as follows:

$$A(s, a) = Q(s, a) - V(s) = r + \gamma V(s') - V(s), \quad (30)$$

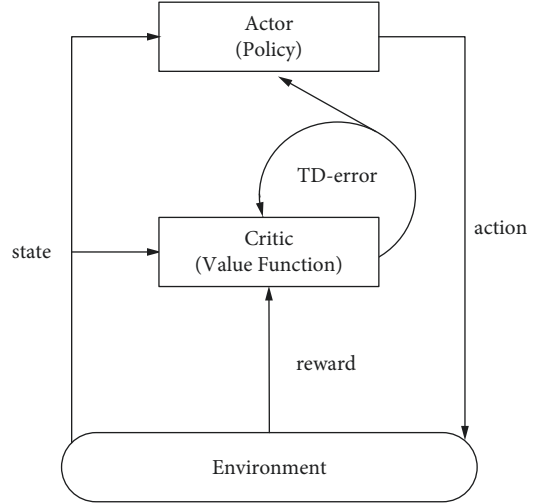


FIGURE 3: Algorithm basic workflow.

where  $A(s, a)$  represents that action  $a$  is good or bad in state  $s$ . If action  $a$  is better than average, then  $A(s, a)$  is positive; otherwise, it is negative.

The algorithm defines the objective function  $J(\pi)$  used to measure the quality of the policy as follows:

$$J(\pi) = E_{\rho^{s_0}}[V(s_0)], \quad (31)$$

where  $J(\pi)$  represents all the average discount rewards obtained by a policy starting from the initial state  $s_0$ .

According to the policy gradient theorem, the algorithm can obtain the definition of the gradient of the objective function:

$$\nabla_{\theta} J(\pi) = E_{s \sim \rho^{\pi}, a \sim \pi(s)}[A(s, a) \cdot \nabla_{\theta} \log \pi(a|s)]. \quad (32)$$

The function obtains the reward obtained from sample  $(s_0, a_0, r_0, s_1)$ , and then the function predicts the value in the next step and provides an estimated approximation. However, the function uses more steps to provide  $n$ -step return.

$$V(s_0) \longrightarrow r_0 + \gamma r_1 + \dots + \gamma^n V(s_n). \quad (33)$$

The advantage of the  $n$ -step return is that the change in the approximate function propagates is faster.

By extending the advantage function  $A(s, a)$ , the gradient  $d\theta$  of the update policy  $\pi$  in the actor network can be obtained as follows:

$$d\theta \longleftarrow d\theta + \nabla_{\theta'} \log \pi(a_t|s_t; \theta') \\ \times \left[ \sum_{l=0}^{k-1} \gamma^l r(t+l) + \gamma^k V(s_{t+k}; \theta'_v) - V(s_t; \theta'_v) \right] + \delta \nabla_{\theta'} H(\pi(s_t; \theta')), \quad (34)$$

where  $H$  represents the entropy to avoid premature convergence to the suboptimal deterministic policy.  $\delta$  is the entropy hyperparameter, which is used to control the strength of the entropy regularization term.  $\theta'_v$  is a parameter of the state value function in the critic network, descending

Input:

- (1) Initialize the actor network with parameter  $\theta$
- (2) Initialize the critic network with parameter  $\theta_v$
- (3) Initialize the actor network parameters  $\theta'_i$  for each thread
- (4) Initialize the critic network parameters  $\theta'_v$  for each thread
- (5) for each SFC  $f \in SF$  do
- (6) for each VSF resource allocation scheme  $x$  in  $X_{n,v}$  in edge cloud do
- (7) if  $x$  is not registered in the VSCB then
- (8) remove  $x$  from  $X_{n,v}$
- (9) end for
- (10) if  $X_{n,v}$  is empty then
- (11) Calculate  $X_{n,v}$  in core cloud
- (12) Select the VSF allocation method  $\theta'_{i,j}$  from  $X_{n,v}$
- (13) Select the VSF allocation method  $\eta'_{i,j,k}$  from  $X_{n,v}$
- (14) Calculate the total cost
- (15) end for
- (16) while the SF set is not empty do
- (17) Set the gradient of the two networks  $d\theta \leftarrow 0$  and  $d\theta_v \leftarrow 0$
- (18) Synchronize the parameters of the thread  $\theta' \leftarrow \theta$  and  $\theta'_v \leftarrow \theta_v$ ,  $t_{\text{start}} = t$
- (19) Obtain state  $s_t$
- (20) Repeat:
- (21) The state features are extracted from the network based on a multiple threshold mechanism, and action  $a_t$  is executed according to policy  $\pi(a_t|s_t; \theta')$
- (22) Get the reward  $r_t$  of environmental feedback and the next state  $s_{t+1}$
- (23)  $t \leftarrow t + 1$
- (24) Until termination status  $S_t$ , or  $t - t_{\text{start}} = t_{\text{max}}$
- (25) Obtain  $R = \begin{cases} 0 & \text{for } \text{terminal } s_t \\ V(s_t, \theta'_v) & \text{for non-terminal } s_t \end{cases}$  from the critic network  
// Boot strap from last state
- (26) for each step  $t \leftarrow t - 1, \dots, t_{\text{start}}$  do
- (27)  $R = r_t + \gamma R$
- (28) Calculate the gradient of the actor network about  $\theta'$ : 
$$d\theta \leftarrow d\theta + \nabla_{\theta'} \log \pi(a_t|s_t; \theta') \times [\sum_{i=0}^{k-1} \gamma^i r(t+i) + \gamma^k V(s_{t+k}; \theta'_v) - V(s_t; \theta'_v)] + \delta \nabla_{\theta'} H(\pi(s_t; \theta'))$$
- (29) Calculate the gradient of the critic network about  $\theta'_v$ : 
$$d\theta_v \leftarrow d\theta_v + \partial (\sum_{i=0}^{k-1} \gamma^i r(t+i) + \gamma^k V(s_{t+k}; \theta'_v) - V(s_t; \theta'_v))^2 / \partial \theta'_v$$
- (30)  $t \leftarrow t - 1$
- (31) end for
- (32) Use  $d\theta$  and  $d\theta_v$  to asynchronously update the global shared parameters  $\theta$  and  $\theta_v$ . Send the  $d\theta$  and  $d\theta_v$  to SFC agent
- (33) end while

**Output:**  $\pi^*(s)$

ALGORITHM 1: RAOA-A3C algorithm.

by a gradient in TD mode. The updated gradient  $d\theta_v$  in the critic network is as follows:

$$d\theta_v \leftarrow d\theta_v + \partial \frac{(\sum_{i=0}^{k-1} \gamma^i r(t+i) + \gamma^k V(s_{t+k}; \theta'_v) - V(s_t; \theta'_v))^2}{\partial \theta'_v} \quad (35)$$

The network structure of the RAOA-A3C algorithm mainly uses convolutional neural networks and fully connected neural networks, and the output of the fully connected layer is used as the input of the actor network and the critic network. The actor network outputs the corresponding

action value to select actions; the critic network outputs a state value to calculate the advantage.

We use Algorithm 1 to solve equation (23). The pseudocode of the RAOA-A3C Algorithm 1 is described as follows:

The RAOA-A3C algorithm consists of two parts. It mainly includes network initialization and resource allocation optimization.  $X_{n,v}$  represents a feasible resource allocation scheme. The mapping service node  $x$  of the VSFs in the edge cloud needs to be authenticated through the blockchain module. If it is not registered on the VSCB,  $x$  will be deleted from the configuration scheme set. Then, when the edge cloud resource configuration scheme is empty,

select the core cloud and authenticate again. VSF resource allocation methods are randomly selected from the available configuration scheme  $X_{n,v}$ . The resources can be optimally allocated.

## 6. Performance Evaluation and Analysis

**6.1. Simulation Settings.** In the experiment, Docker 18.06, Python 3.0, TensorFlow, and OpenAI Gym were installed in Ubuntu 16.04 to configure the environment, and MATLAB was used for simulation experiments. Other related settings are as follows: the virtual machines are interconnected via a 1Mbit/s virtual LAN card, and each blockchain node has a 2.0 GHz 8-VCPUs attribute. There are 30 nodes (10 server nodes and 20 switches) and 50 links. The discount factor  $\gamma$  is 0.9, and the entropy hyperparameter  $\delta$  is 0.1 [10]. The maximum delay limit of SFC  $D_i$  is 30 ms; the data packet arrival process follows the independent identical distributed Poisson process; and the parameter value is  $\lambda_i = 2$ . The packet size is 500 kb/packet; the physical link bandwidth resource is 640 MB; the CPU resource capacity of server  $n$  is 8 cores; the service rate of a single CPU is  $\ell = 25MB/s$ ; the positive number is  $\alpha = 30$ ; and the positive number  $\beta = 20$  [38].

**6.2. Performance Evaluation.** Firstly, eight and ten consortium peers were deployed on the core cloud and the edge cloud, respectively; for comparison, it is verified that the RAOA-A3C algorithm has the performance of consistent delay.

As shown in Figure 4, the consensus efficiency of the core cloud is higher than the edge cloud. The delay increases significantly as the number of SFCs increases. The reason is that users who send service requests need to be authenticated; SFC transactions also need to be registered on the VSCB; and when the number of consortium partners increases, the consensus delay also increases with the increase in the number of SFCs.

The weighted values  $e1$  and  $e2$  are iterated by using max equality constraints, optimality constraints, and max inequality constraints. Assuming that 200 SFCs arrive in the virtual network, and after 10,000 iterations, the allocation cost and average delay are shown in Figures 5 and 6.

As shown in Figures 5 and 6, after 10,000 iterations, the allocation cost and the average delay under different constraints are obtained. When the number of iterations reaches about 6,000, the convergence is obvious. The algorithm is effective.

In the following, the RAOA-A3C algorithm is compared with DQN [21] and A3C [10] algorithms in the three aspects of total allocation cost, average delay, and utility function.

As shown in Figures 7 and 8, as the number of SFCs increases, the average delay and total allocation cost of the three algorithms are increasing. The RAOA-A3C algorithm has a lower allocation cost than DQN and A3C algorithms. DQN algorithm is mainly suitable for solving the discrete action space, but the action space in this paper is a continuous value, which causes the DQN algorithm to be

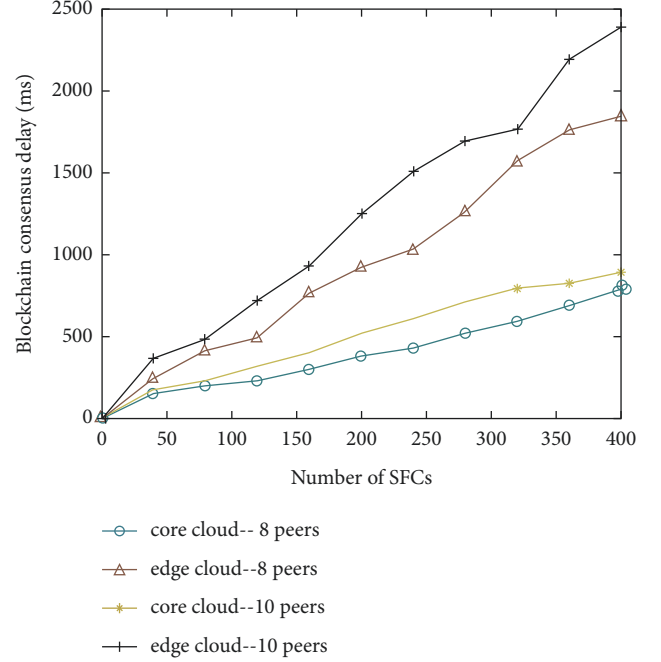


FIGURE 4: Blockchain conformance delay.

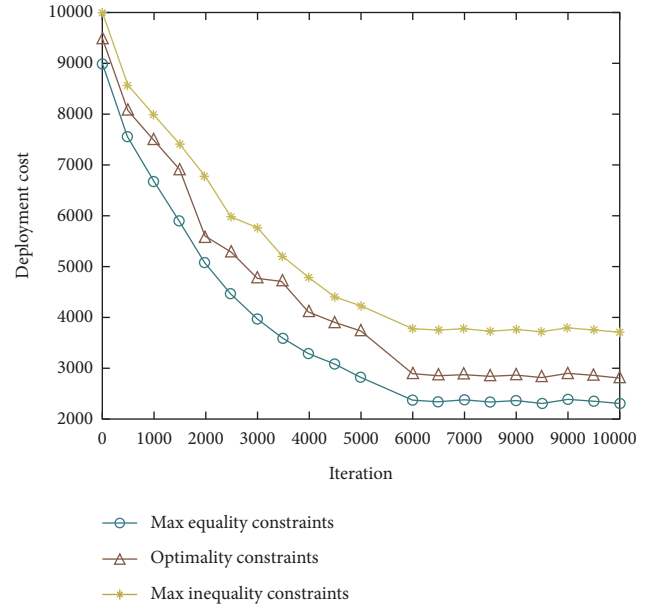


FIGURE 5: Comparison of allocation cost under different constraints.

significantly weaker than the A3C algorithm and the RAOA-A3C algorithm in optimizing the delay and allocation cost.

As shown in Figure 8, when the number of SFCs is less than 100, the RAOA-A3C algorithm is higher than the A3C algorithm in the system average delay. The reason is that there is a time delay in the resource authentication stage. However, as the number of SFCs increases, the proportion of time delay in the blockchain becomes smaller, and the influence becomes weaker, and advantages of the RAOA-A3C algorithm are revealed.



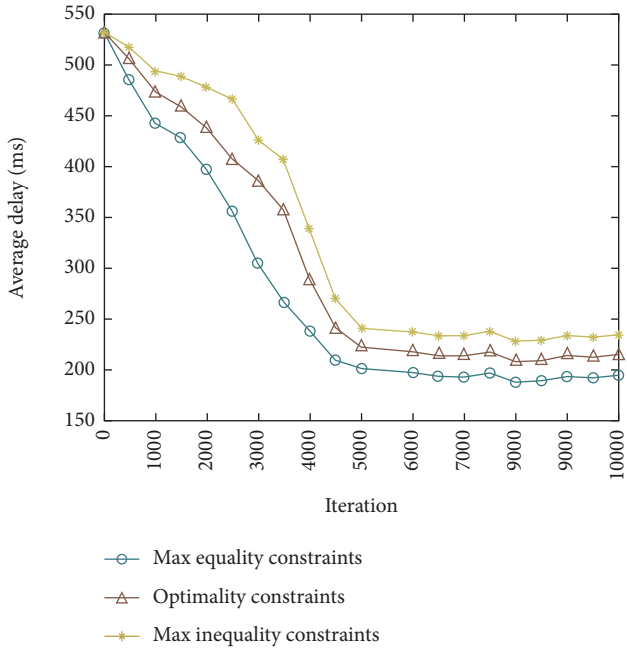


FIGURE 6: Comparison of average time delays under different constraints.

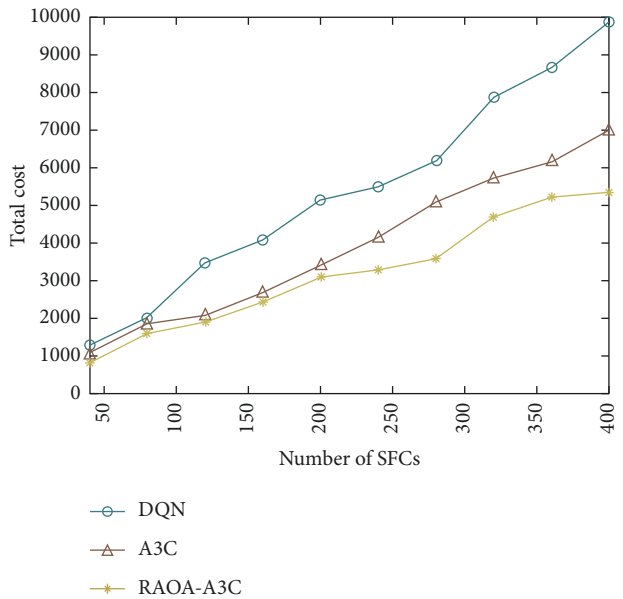


FIGURE 7: Total cost of allocation comparison.

As shown in Figure 9, the total system delay and total allocation cost increase; as the number of SFCs increases, the utility will also decrease as the number of SFCs increases, but the utility of the RAOA-A3C algorithm proposed in this paper decreases the slowest.

As shown in Figure 10, when the number of SFCs is less than 200, the RAOA-A3C algorithm fluctuates due to the influence of the consortium blockchain. When the number of SFCs is greater than 200, the advantages of RAOA-A3C

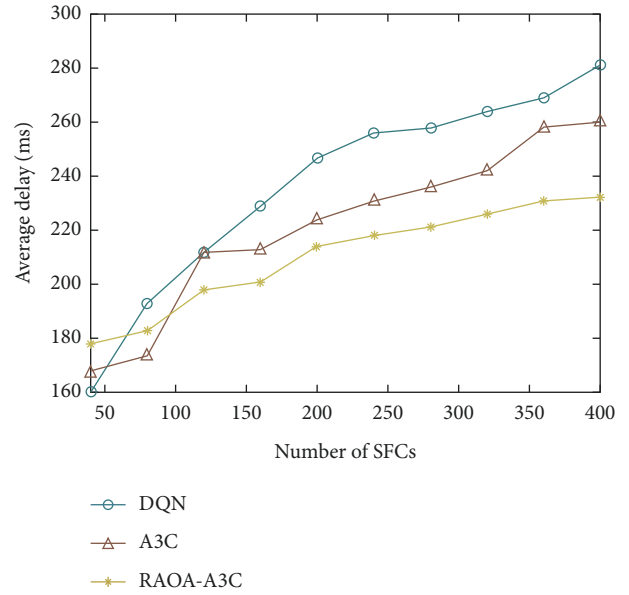


FIGURE 8: Average delay comparison.

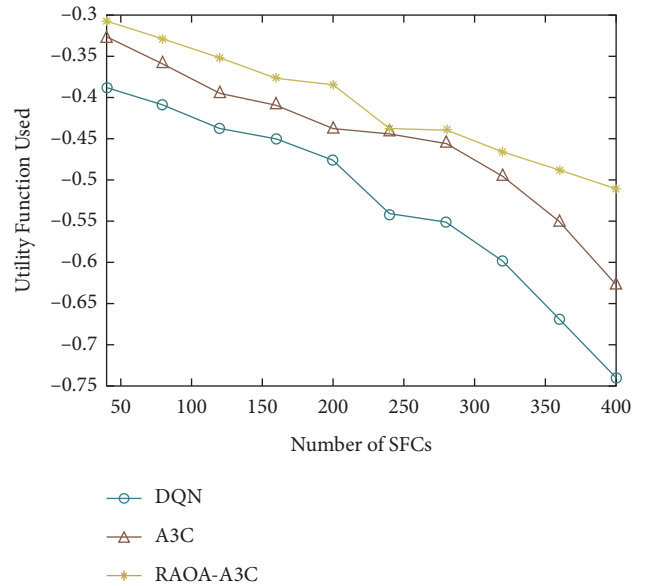


FIGURE 9: Utility function comparison.

are better reflected, and the variance of server usage of RAOA-A3C is lower than that of DQN and A3C.

As shown in Figure 11, the variance of the utilization rate of the link of RAOA-A3C is lower than that of DQN and A3C on the whole.

As shown in Figures 10 and 11, the experiment compares the variance of the link usage rate and the variance of the server usage rate of the DQN, A3C, and RAOA-A3C algorithms. It can be seen that a smaller difference indicates that the service is more evenly distributed on the server and the link. The RAOA-A3C algorithm congestion control results are better.

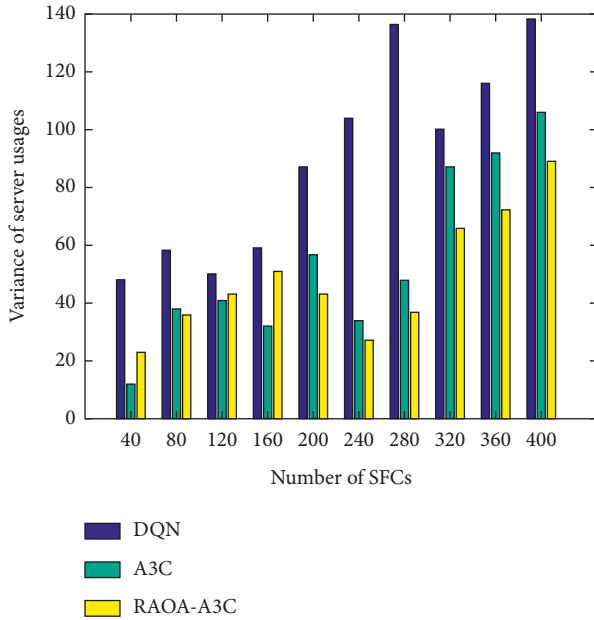


FIGURE 10: Comparison of server usage under different numbers of SFCs.

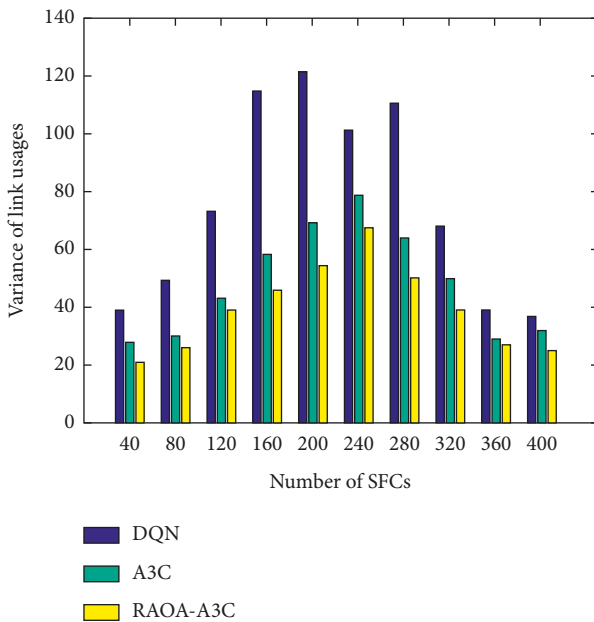


FIGURE 11: Comparison of link usage under different numbers of SFCs.

## 7. Conclusion

Although the power video monitoring system based on the cloud-edge computing architecture brings many benefits. The problems of distribution and heterogeneity cannot guarantee the reliability and durability of the service. In order to establish trust between service providers and users, VSCB is integrated into the management of the power video monitoring system. In addition, the SFV technology was first proposed to realize the optimal allocation of resources; the

MDP model was constructed; and then the RAOA-A3C algorithm was proposed. Simulation experimental results show the advantages of the resource allocation optimization model in cost-saving, time-saving, and security.

## Data Availability

No data were used to support this study.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research was funded by the Provincial Science and Technology Innovation Special Fund Project of Jilin Province, grant no. 20190302026GX; Natural Science Foundation of Jilin Province, grant no. 20200201037JC; the Higher Education Research Project of Jilin Association for Higher Education, grant no. JGJX2018D10; the Fundamental Research Funds for the Central Universities for JLU, Platform of Jilin Province Science and Technology Department, grant no. 20190902011TC; 2021 Digital transformation and innovation platform construction project of Jilin Provincial Development and Reform Commission, grant nos. 2021C049; 2020 Industrial Technology Research and Development Project of Jilin Development and Reform Commission, grant no. 2020C020-1; 2019 Jilin Province S&T Development Plan Technology Research Project Research Project, grant no. 20190302115GX; Changchun Philosophy and Social Science Planning Project, grant no. CSKT2021ZX-054; and Changchun Institute of Technology Science and Technology Fund Project, grant no. 320200010.

## References

- [1] Y. Y. Sun, T. Jia Zhen, and H. Y. Zhu, "A review of multimodal deep learning," *Computer Engineering and Applications*, vol. 56, no. 21, pp. 1-10, 2020.
- [2] B. Wang, B. H. Wang, R. Zhang, and F. J. Li, "Research on the security architecture of power wireless communication system based on TD-LTE," *Microcomputer Applications*, vol. 36, no. 7, pp. 61-63, 2020.
- [3] S. Zhang, J. Zhang, Y. Wang, J. L. Liu, B. Yan, and Z. W. Shang, "Research on pedestrian Re-identification network architecture technology based on multi-view and cross-modality in power field," *Sichuan Power Technology*, vol. 43, no. 6, pp. 6-10+15, 2020.
- [4] L. Yang, M. Li, X. Y. Ye, E. C. Sun, and Y. H. Zhang, "Research on optimal allocation of industrial Internet resources by integrating edge computing and blockchain," *High Technology Communications*, vol. 30, no. 12, pp. 1253-1263, 2020.
- [5] P. Yu, H. Yan, H. L. Chen, and D. Long, "Joint optimization of collaborative edge caching and resource allocation for mobile terminal energy saving based on edge computing," *Test Engineering and Management*, vol. 83, no. 5-6, Article ID 18994, 2020.
- [6] Aliyun Computing Co Ltd, "China Institute of standardization of electronic technology," *White Paper on Edge Cloud Computing Technology and Standardization*, 2018.

- [7] M. S. Munir, S. F. Abedin, N. H. Tran, and C. S. Hong, "When edge computing meets microgrid: a deep reinforcement learning approach," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 7360–7374, 2019.
- [8] M. Crosby and P. N. Pattanayak, "Blockchain technology: beyond bitcoin," *Applied Innovation Review*, vol. 2, pp. 6–19, 2016.
- [9] Y. Sun, S. Yang, G. J. Gong, J. X. Yang, and B. Zhou, "Research on endogenous security of power distribution Internet of things based on trusted computing and blockchain," *Huadian Technology*, vol. 42, no. 8, pp. 61–67, 2020.
- [10] S. Guo, Y. Dai, S. Xu, and F. Qi, "Trusted cloud-edge network resource management: DRL-driven service function chain orchestration for IoT," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6010–6022, 2020.
- [11] X. Y. Zhang, C. S. Li, H. C. Shi, L. Peng, and D. Jing, "AdapNet: adaptability decomposing encoder-decoder network for weakly supervised action recognition and localization," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2020.
- [12] X. Y. Zhang, S. P. Wang, and X. C. Yun, "Bidirectional active learning: a two-way exploration into unlabeled and labeled dataset," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 12, pp. 3034–3044, 2015.
- [13] X. Y. Zhang, H. C. Shi, C. S. Li, and L. Peng, "Multi-instance multi-label action recognition and localization based on spatio-temporal pre-trimming for untrimmed videos," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pp. 12886–12893, 2020.
- [14] D. B. Zhao, K. Shao, Y. H. Zhu et al., "A review of deep reinforcement learning: concurrently discuss the development of computer go," *Control Theory & Applications*, vol. 33, no. 6, pp. 701–717, 2016.
- [15] J. G. Wu, T. L. Liu, J. Y. Li, and J. Y. Huang, "Research progress of blockchain technology in mobile edge computing," *Computer Engineering*, vol. 46, no. 8, pp. 1–13, 2020.
- [16] C. Li, X. D. Duan, and W. Chen, "Thoughts and practices about SDN and NFV," *Telecommunications Science*, vol. 30, no. 8, pp. 23–27, 2014.
- [17] M. Jin, L. L. Li, W. J. Zhang, and W. Liu, "Service function chain mapping algorithm based on deep reinforcement learning," *Application Research of Computers*, vol. 37, no. 11, pp. 3456–3460+3466, 2020.
- [18] W. H. Zhan, J. Wang, Q. X. Zhu, H. C. Duan, and Y. L. Ye, "Computational offloading scheduling method based on deep reinforcement learning in mobile edge computing," *Application Research of Computers*, vol. 38, no. 1, pp. 241–245+263, 2021.
- [19] D. Silver, T. Hubert, J. Schrittwieser et al., "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018.
- [20] V. Mnih, K. Kavukcuoglu, D. Silver, A. Anderj, J. Veness, and A. Graves, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [21] D. P. Wu, Q. R. Liu, H. G. Wang, W. Dalei, and W. Ruyan, "Socially aware energy-efficient mobile edge collaboration for video distribution," *IEEE Transactions on Multimedia*, vol. 19, no. 10, pp. 2197–2209, 2017.
- [22] S. Z. Bi and Y. J. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 4177–4190, 2018.
- [23] L. X. Chen, S. Zhou, and J. Xu, "Computation peer offloading for energy-constrained mobile edge computing in small-cell networks," *IEEE/ACM Transactions on Networking*, vol. 26, no. 4, pp. 1619–1632, 2018.
- [24] Y. J. Zhu, "A survey on mobile edge platform with blockchain," in *Proceedings of the 3rd Information Technology, Networking, Electronic and Automation Control Conference*, pp. 879–883, IEEE Press, Washington D.C., USA, March 2019.
- [25] H. Jiang, F. Shen, S. Chen, and K.-C. Li, "A secure and scalable storage system for aggregate data in IoT[J]," *Future Generation Computer Systems*, vol. 49, pp. 133–141, 2015.
- [26] G. Oliva, S. Cioaba, and C. N. Hadjicostis, "Distributed calculation of edge-disjoint spanning trees for robustifying distributed algorithms against man-in-the-middle attacks," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 4, pp. 1646–1656, 2018.
- [27] J. J. Fang and K. Lei, "Summary of blockchain technology for edge artificial intelligence computing," *Journal of Applied Sciences*, vol. 38, no. 1, pp. 1–21, 2020.
- [28] W. Tang, X. Zhao, W. Rafique, and W. Dou, "A blockchain-based offloading approach in fog computing environment," in *Proceedings of the IEEE Intl Conf on Ubiquitous Computing & Communications; IEEE Intl Conf on Parallel & Distributed Processing with Applications; IEEE Intl Conf on Social Computing & Networking; IEEE Intl Conf on Big Data & Cloud Computing; IEEE Intl Conf on Sustainable Computing & Communications*, Melbourne, VIC, Australia, December 2018.
- [29] Z. F. Ma, X. C. Wang, D. K. Jain, H. Khan, G. Hongmin, and W. Zhen, "A blockchain-based trusted data management scheme in edge computing," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 2013–2021, 2020.
- [30] A. Jindal, G. S. Aujla, and N. Kumar, "SURVIVOR: a blockchain based edge-as-a-service framework for secure energy trading in SDN-enabled vehicle-to-grid environment," *Computer Networks*, vol. 153, pp. 36–48, 2019.
- [31] M. Samaniego and R. Deters, "Hosting virtual IoT resources on edge-hosts with blockchain," in *Proceedings of the IEEE International Conference on Computer and Information Technology (CIT)*, pp. 116–119, Nadi, Fiji, December 2016.
- [32] M. Samaniego and R. Deters, "Virtual resources & blockchain for configuration management in IoT," *Journal of Ubiquitous Systems and Pervasive Networks*, vol. 9, no. 2, pp. 1–13, 2017.
- [33] P. Vizarrreta, M. Condoluci, and C. M. Machuca, "QoS-driven function placement reducing expenditures in NFV deployments," in *Proceedings of the IEEE International Conference on Communications*, pp. 1–7, IEEE Press, Paris, France, May 2017.
- [34] G. Xiong, Y. X. Hu, L. Tian, and J.-L. Lan, "A virtual service placement approach based on improved quantum genetic algorithm," *Frontiers of Information Technology & Electronic Engineering*, vol. 17, no. 7, pp. 661–671, 2016.
- [35] J. Sun, G. Y. Zhu, G. Sun, and D. Liao, "A reliability-aware approach for resource efficient virtual network function deployment," *IEEE Access*, vol. 12, no. 6, Article ID 18238, 2018.
- [36] C. Qiu, F. R. Yu, H. P. Yao, and C. Zhano, "Blockchain based software-defined industrial Internet of Things: a dueling deep Q-learning approach," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4627–4639, 2019.
- [37] W. H. Zhan, C. B. Luo, J. Wang, G. Min, and H. Duan, "Deep reinforcement learning-based computation offloading in vehicular edge computing," in *Proceedings of the IEEE Global Communications Conference*, pp. 1–6, IEEE Press, Piscataway, NJ, USA, December 2019.

- [38] L. Q. He, *Research on Virtual Network Function Deployment and Migration Optimization Algorithm Based on Deep Reinforcement Learning*, Chongqing University of Posts and Telecommunications, Chongqing, China, 2020.
- [39] R. S. Sutton and A. G. Barto, *Reinforcement Learning: an Introduction*, pp. 51–85, The MIT Press, Massachusetts, MA, USA, 1998.
- [40] T. W. Kuo, B. H. Liou, and C. J. Lin, “Deploying chains of virtual network functions: On the relation between link and server usage,” *IEEE/ACM Transactions on Networking*, vol. 26, no. 4, pp. 1562–1576, 2018.
- [41] Z. L. Ye, X. J. Cao, J. P. Wang, H. Yu, and C. Qiao, “Joint topology design and mapping of service function chains for efficient, scalable, and reliable network functions virtualization,” *IEEE Network*, vol. 30, no. 3, pp. 81–87, 2016.
- [42] M. Otokura, K. Leibnitz, Y. Koizumi, D. Kominami, T. Shimokawa, and M. Murata, “Evolvable virtual network function placement method: Mechanism and performance evaluation,” *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 1, pp. 27–40, 2019.
- [43] X. Han, Y. Yuan, and F. Y. Wang, “Security problems on blockchain: the state of the art and future trends,” *Acta Automatica Sinica*, vol. 45, no. 1, pp. 206–225, 2019.
- [44] A. M. Antonopoulos, *Mastering Blockchain Programming the Open Blockchain*, O’Reilly Media, Massachusetts, MA, USA, 2017.
- [45] J. Gao, X. G. Yan, B. Liang, and H. Guo, “Research on IIoT architecture of edge computing based on blockchain,” *Application Research of Computers*, vol. 37, no. 07, pp. 2160–2166, 2020.
- [46] Ministry of Industry and Information Technology, *China Blockchain Technology and Application Development White Paper*, 2016.
- [47] L. Tang, H. Yang, G. F. Zhao, Y. W. Wang, and Q. B. Chen, “5G network slicing node and link mapping algorithm based on delay perception,” *Journal of Beijing University of Posts and Telecommunications*, vol. 41, no. 6, pp. 71–77, 2018.
- [48] A. L. Cheng, J. Li, Y. L. Yu, and H. Jin, “Delay-sensitive user scheduling and power control in heterogeneous networks,” *IET Networks*, vol. 4, no. 3, pp. 175–184, 2015.
- [49] J. Yang, S. B. Zhang, X. M. Wu, and Y. Ran, “Online learning-based server provisioning for electricity cost reduction in data center,” *IEEE Transactions on Control Systems Technology*, vol. 25, no. 3, pp. 1044–1051, 2017.