Research Article

Research on Intelligent Mapping Algorithm of Secure Virtual Network under Cloud Computing

Xiaoqiang Niu, Hao Lan, and Dan Zeng

Gannan Health Vocational College, 341000, China

Correspondence should be addressed to Xiaoqiang Niu; niu_xiaoqiang@gnhvc.edu.cn

Received 1 April 2022; Revised 18 April 2022; Accepted 23 April 2022; Published 7 May 2022

1. Introduction

As a milestone technology, cloud computing has greatly promoted the development of information society, and cloud computing has achieved rapid development. Google, Amazon, Microsoft, IBM, Alibaba, and other large companies have put forward their own cloud computing platforms, and the research on various related technologies of cloud computing has also shown a trend of flowering [1]. To put it simply, cloud computing is a parallel computing model based on the Internet. In a data center, thousands of servers connect to a physical resource pool. Users can access the computing resources and complete their own computing tasks. Cloud computing center is essentially a large-scale server cluster, which provides powerful computing capacity and improves its resource utilization. It is one of the pursuits of cloud computing providers. Network virtualization is one of the key technologies for the development of the Internet in the future. Its purpose is to overcome the chronic problems in the current Internet and promote the development of the Internet [2, 3]. In network virtualization, a major challenge is how to construct a virtual network in the underlying network, or how to allocate resources to the virtual network is an important problem, often referred to as the virtual network mapping problem [4].

As one of the core technologies of cloud computing, virtual network technology greatly improves the effective utilization of physical resources. With the continuous development of virtualization network, its characteristics of environment customization, isolation, and security are gradually improved. In the past decade, a large number of heuristic algorithms have emerged, some of which have solved the problem of virtual network resource allocation in a relatively mature way [5]. A better mapping solution can not only improve the efficiency of virtual network operation, but also facilitate the completion of parallel jobs. It also reduces the amount of electricity used in data centers and the heat generated by servers, thereby saving infrastructure providers money. In a data center, the entire underlying topology is likely to change. Searching for a better virtual network resource redistribution algorithm that supports the change of the underlying topology can further improve the adaptability and survivability of the virtual network, improve the service quality of cloud computing, and provide better services for users. With the development of technology, cloud computing users can enjoy better services while using lower...
prices. From the perspective of The Times, the improvement of resource utilization rate is also in line with the pace of the current society, to achieve the purpose of green computing, in line with the concept of "green IT."

Therefore, Rong put forward, based on the cost and power consumption of joint optimization of SDN virtual network mapping algorithm [5], the mapping of virtual node and link to evaluate the cost and power consumption, on the basis of modeling VNE costs and the cost of power function, and meet the demand of resource constraints such as modeling the VNE model based on minimizing the total cost function. The optimization problem as an integer linear programming problem, difficult to solve directly, batch virtual network mapping strategy based on time window dynamic processing request online, and then to a specific time window VNR and transform it into a virtual node mapping subproblem and virtual link subproblems, and application of heuristic algorithms to solve two subproblems, which determine the VNR mapping strategy, this algorithm effectively reduces the energy cost, but the mapping efficiency is not good. Gang et al., based on the virtual software-defined network mapping algorithm of topology segmentation and cluster analysis [6], reduced the complexity of physical network by topology segmentation based on the shortest hop number. Through cluster analysis based on node topology and resource attributes, the link constraints are distributed to node bandwidth resources, and the degree of node constraints are considered, and the nodes that do not meet the link requirements are remapped, so as to optimize the node-link mapping process. This algorithm effectively improves the mapping efficiency, but the node energy consumption is too large. In view of the above problems, this paper designs the intelligent mapping algorithm of secure virtual network under cloud computing, puts forward the example diagram of network virtualization according to the network virtualization technology, then constructs the energy consumption model and network load model of virtual network mapping problem, analyzes the mapping reliability, and finally determines the virtual network mapping strategy through the optimization of cloud computing technology to realize the research on the intelligent mapping algorithm of secure virtual network under cloud computing. The experimental results show that the percentage of nodes used in this algorithm is far lower than other compared mapping algorithms, and the average energy consumption is far lower than other methods, which can effectively reduce the complexity of network mapping and improve the efficiency of network mapping.

2. Network Virtualization Technology

The original network structure followed the principle of simple access, but the development of network structure has become very complex, and the construction of new network architecture has become extremely difficult. As a new technology, network virtualization has gradually become one of the key technologies to solve the problem of network rigidity.

As shown in Figure 1, network virtualization technology allows the underlying infrastructure to be abstracted into a unified programming interface using virtualization technology on top of the original network architecture, mapping multiple completely isolated virtual networks with completely different topologies onto the same underlying infrastructure [7]. Virtualization technology provides a virtualization view of resources being used to instantiate virtual machines (VMs). Virtual monitor devices or hypervisors manage isolated virtual machines to control multiple access to physical resources. The multivariate theory of network virtualization stipulates that networks are fully virtualized and services are provided independently by potential communication infrastructure providers [8].

Figure 2 shows a two-tier model of virtualization technology, where infrastructure providers (InPs) are responsible for allocating physical network resources to service providers (SPs), including CPU, memory, and link resources of physical network nodes. Service providers (SPs) are responsible for providing services externally using the resources allocated to them [9–12]. This model makes the virtual network with node and link constraints act on the infrastructure. The underlying infrastructure directly allocates a physical node to each virtual node requested by the virtual network and a physical network path or link to each virtual link. Obviously, a virtual network can be mapped onto one or more physical networks. This approach allows infrastructure providers to independently manage their own underlying infrastructure, while service providers focus on providing E2E services to end users [13].

Virtualization technology’s four-tier business model to describe, request, create, and manage virtual networks. As shown in Figure 3, the model is composed of four roles: infrastructure providers (InPs), virtual network provider (VNP), virtual network operator (VNO) and service providers (SPs) [14–17]. SPs request resources from the VNP to provide services to users. The VNP is responsible for creating virtual networks based on SPs requests, while the VNO is responsible for managing the virtual networks that are successfully established. VNP and VNO separate SPs from the task of creating and managing virtual networks in a two-tier model to focus on providing services to end users.

Network virtualization technologies have evolved from virtual LANS (VLANs), VPNS, active programming networks, and overlay networks to multiple network theories, which enable multiple and heterogeneous physical networks to provide flexible and scalable cloud computing services. Based on the concept of network virtualization and virtualization technologies, traffic loads and communication operations can be easily controlled and managed to better support on-demand parallel processing systems for big data applications, online gaming, high definition television (HDTV), and on-demand bandwidth, such as MapReduce [18, 19]. Two virtual nodes from the same virtual network request can be mapped to two different physical nodes, while two virtual nodes from different virtual network requests can be mapped to the same physical node, and the main resources of the virtual node and virtual link request are CPU and bandwidth, respectively. Virtual network technology has
the advantages of fast and convenient network construction, reducing network investment, saving use cost, network security and reliability, and simplifying users’ maintenance and management of the network. Many device-based virtualization providers also provide additional functional modules to improve the overall performance of secure virtual network intelligent mapping algorithm, which can obtain better performance and more perfect functions than the standard operating system.

### 3. Intelligent Mapping Algorithm for Secure Virtual Network in Cloud Computing

#### 3.1. Energy Consumption Model of Virtual Network Mapping

Based on the existing research results of energy saving virtual network and energy consumption models of various network components, considering time cost and environmental factors, this paper establishes the energy consumption model of multidomain virtual network mapping problem. The energy consumption model consists of three parts:

The energy consumption model for virtual network mapping is established in an environment where all physical nodes in the entire multidomain physical network $G_{whole}$ remain closed until the first virtual network mapping (for example, $G^1$) is executed. After performing the node mapping of the first virtual network, multiple physical nodes will be activated to a smooth working state. In this section, constraints are considered for mapping each virtual node: CPU, node storage, node location, and node processing time [20]. Node storage and node processing time have little effect on the energy cost of active/nonactive physical nodes. The effect of node location on energy consumption of physical nodes will be deferred to the third part of this section [21, 22]. Therefore, this section focuses on the impact of CPUS on power consumption of physical nodes. In this paper, the well-known Server Energy Cost Model (Server Energy Cost Model) is adopted to quantify the CPU Energy consumption of nodes to physical network.

CPU utilization dominates the energy consumption variation of mapped physical nodes [23–25]. If an underlying physical node is not mapped by any virtual node, the energy consumption of the physical node is 0. See the lower part of equation (1) for details. This section uses physical node $M^p$ as an example.

Node $M^p$ is closed at the beginning [26, 27]. After the virtual network mapping is completed, $M^p$ is mapped by the virtual node $M^v$. See formula (1) for detailed energy consumption.

$$P(M^p) = \begin{cases} P_b + P_1 \cdot CPU(M^v), & \text{Node } M^p \text{ works for the first time,} \\ 0, & \text{close.} \end{cases}$$

(1)

$P_b$ is the basic energy consumption of physical node $M^p$. $P_{max}$ represents the total energy consumption of a physical node at maximum CPU utilization [28]. $P_1 = P_{max} - P_b$, $P_1$ indicates the CPU usage factor of a physical node. The mapped physical node must have sufficient CPU resources to accommodate the mapped virtual node $M^v$. For the remaining virtual nodes in the first virtual network, formula (1) is adopted in this paper to measure the node energy. 

---

**Figure 1:** Network virtualization example diagram.

**Figure 2:** Two-tier model of virtualization technology.
consumption of the virtual network [29]. The reason for this is that the multidomain physical network $G_{\text{whole}}$ is shut down. All physical components are turned off.

After the first virtual network mapping is completed, physical nodes that are partially powered coexist with physical nodes that are shut down. Then, the remaining virtual network mapping is carried out [30–32]. The detailed energy consumption of the remaining virtual network mapping is shown in formula (2). If the mapped physical node is initially closed, the energy consumption is shown in equation (1).

$$P(M^p) = \begin{cases} P_b + P_1 \cdot CPU(M^p), & \text{Node } M^p \text{ works for the first time}, \\ P_1 \cdot CPU(M^p), & \text{Node } M^p \text{ is in power supply operation state}, \\ 0, & \text{closed}. \end{cases}$$

(2)

In the energy consumption of virtual link research, no matter the first $G^v$ or any virtual network, each virtual link will be mapped to a separate physical path [33].

A mapped physical path can consist of one or more directly connected physical links. In other words, there are multiple transit physical nodes on the physical path mapping this virtual link. These terminal physical nodes, including source nodes and end nodes, are responsible for forwarding or receiving packets [34]. The energy consumption of such nodes (all expressed as $P'(M^p)$) is shown as

$$P'(M^p) = \begin{cases} P'_b + Num \cdot P_{\text{con}}, & \text{Node } M^p \text{ works for the first time}, \\ P\cdot CPU(M^p), & \text{Node } M^p \text{ is in power supply operation state}, \\ 0, & \text{The node is in the mapped physical path}, \end{cases}$$

(3)

where $P'_b$ refers to the basic energy consumption of the transit node. $P'_b$ is different from $P_b$ in equations (1) and (2). As for $P_{\text{con}}$, it is used to measure the energy consumption of the transit node that forwards or receives data packets [35]. Whether the node $M^p$ remains idle or full, the value of $P'_b$ remains constant. For simplicity, this section sets the same $P_{\text{con}}$ value for all transit nodes. The variable $Num$ is used to calculate the number of times the transit node forwards or receives packets. In the research of multidomain virtual network mapping, the whole underlying multidomain physical network is composed of several different physical networks distributed by region. These physical networks are distributed in different geographical locations. Therefore, researchers must consider the impact of the location of physical network distribution on the energy cost of virtual network mapping. This section considers the power expenditure used to calculate the energy consumption of virtual network mapping. In actual production and life, electricity charges in different areas are different. Therefore, in the long run, the energy price of different regions will affect the mapping income and mapping energy consumption.

In this context, this paper investigates energy prices (electricity prices) in major Chinese cities. Five major cities are selected in this paper: B, S, N, C, and X. Figure 4 is drawn in this paper. Figure 4 records hourly electricity prices in five cities since June 2020.

Obviously, electricity prices are different in different cities/regions. Even within the same city/region, electricity prices can change over time. In order to quantify the impact of electricity price attributes on virtual network mapping energy consumption, this paper adopts the duration simulation mode (see simulation experiment Settings).

$$EnCo(G^v) = Pri(t) \cdot LifeTime(G^v) \cdot \sum_{M^p \in M^{0}} \sum_{M^v \in M^{0}} X_{M^v}^{M^p} \cdot P(M^p) +,$$

$$Pri(t) \cdot LifeTime(G^v) \cdot \sum_{M^p \in M^{0}} \sum_{M^v \in M^{0}} X_{M^v}^{M^p} \cdot P'(M^p).$$

(4)
Pri(t) represents the electricity price in any hour \( t \in [0, 23] \). The definition of binary variable \( X \) is the same as that in the previous chapter. As shown in Figure 4, the electricity prices of the five cities show different states at different stages. In order to ensure the load balance of the network, different virtual nodes in the same model will be mapped to different physical nodes, especially when the virtual nodes have no location requirements, which is particularly important to ensure that a better mapping scheme can be found in a short execution time and the global optimal solution can be obtained.

3.2. Network Load Modeling. The network load modeling environment needs to comprehensively consider the corresponding network load of \( k \) virtual networks mapped to the underlying network, the modeling network load \( \Psi \) is

\[
\Psi = \sum_{k=1}^{K} \Psi_k
\]

In equation (5), \( \Psi_k \) represents the network load mapped from \( G_k^V \) to \( G^V \).

Among them,

\[
\Psi_k = \Psi_{k}^n + \Psi_{k}^e
\]

where \( \Psi_{k}^n \) represents the node load mapped from \( G_k^V \) to \( G^V \). \( \Psi_{k}^e \) indicates the link load mapped from \( G_k^V \) to \( G^V \).

In equation (6), \( \Psi_{k}^n \) is expressed as

\[
\Psi_{k}^n = \sum_{n'_{k,i} \in N_k^V, n_i' \in N^V} \alpha_{k,i} \left( \eta_C \frac{C(n'_{k,i})}{C_k(n'_{k,i})} + \eta_S S(n'_{k,i}) + \eta_T T_k(n'_{k,i}) \right)
\]

In equation (7), \( \alpha_{k,i} \) represent node mapping identifier, and \( \alpha_{k,i} \in \{0, 1\} \), \( \alpha_{k,i} = 1 \) represent \( n'_{k,i} \) mapping to \( n_i' \).

Otherwise, \( \alpha_{k,i} = 0 \), \( \eta_C, \eta_S \), and \( \eta_T \) represent the weight factor, \( \eta_C + \eta_S + \eta_T = 1 \), \( \epsilon \) represents the minimum positive number, and avoid the denominator being zero. \( C_k(n'_{i}) \), \( S_k(n'_{i}) \), and \( T_k(n'_{i}) \) represent the remaining CPU capacity, storage capacity, and TCAM capacity of \( n_i' \) after the first \( k-1 \) virtual network mapping.

In equation (7), \( C_k(n'_{i}) \) is expressed as

\[
C_k(n'_{i}) = C(n'_{i}) - \sum_{j=1}^{k-1} \sum_{n'_{j,i} \in N_j^V} \alpha_{j,i} C(n'_{j,i})
\]

In equation (7), \( S_k(n'_{i}) \) is expressed as

\[
S_k(n'_{i}) = S(n'_{i}) - \sum_{j=1}^{k-1} \sum_{n'_{j,i} \in N_j^V} \alpha_{j,i} S(n'_{j,i})
\]
In equation (7), $T_k(n'_i)$ in $n$ is expressed as

$$T_k(n'_i) = T(n'_i) - \sum_{l=1}^{k-1} \sum_{e \in N'_i} a_{l,u,i} T(n''_i) - \sum_{l=1}^{k-1} \sum_{e \in E'_i} \beta_{l,u,i} T(e''_{l,x,r}).$$

(10)

In equation (10), $\beta_{l,u,i}$ represents the intermediate node mapping identifier, $\beta_{l,u,i} \in \{0, 1\}$, and $\beta_{l,u,i} = 1$ represents that $n'_i$ is the intermediate node of the underlying path carrying $e''_{l,x,r}$. Otherwise $\beta_{l,u,i} = 0$. $T(e''_{l,x,r})$ represents the TCAM capacity consumption of the intermediate node carrying the underlying path of $e''_{l,x,r}$.

In equation (10), $T(e''_{l,x,r})$ is expressed as

$$T(e''_{l,x,r}) = \theta(T(n''_i) + T(n''_i)).$$

(11)

In equation (11), $\theta$ represents the TCAM capacity consumption factor of the intermediate node, $\theta \in (0, 0.5)$. In equation (12), $\phi_k^e$ is expressed as

$$\phi_k^e = \sum_{e \in E'_i} \sum_{e \in E'_i} \gamma_{k,u,i,j} \frac{B(e''_{l,x,r})}{B_k(e''_{l,x,r})} + \epsilon,$$

(12)

where $\gamma_{k,u,i,j}$ represents the link mapping identifier, $\gamma_{k,u,i,j} \in \{0, 1\}$, and $\gamma_{k,u,i,j} = 1$ represents that $e''_{k,u,i}$ is mapped to $e''_{l,x,r}$. Otherwise, $\gamma_{k,u,i,j} = 0$.

$B_k(e''_{l,x,r})$ represents the remaining bandwidth capacity of $e''_{l,x,r}$ after the first $k - 1$ virtual network mapping.

In equation (12), $B_k(e''_{l,x,r})$ is expressed as

$$B_k(e''_{l,x,r}) = B(e''_{l,x,r}) - \sum_{l=1}^{k-1} \sum_{e \in E'_i} \gamma_{l,u,i,j} B(e''_{l,x,r}).$$

(13)

3.3. Mapping Reliability Analysis. Comprehensively consider the mapping reliability corresponding to $k$ virtual networks mapped to the underlying network, and model the mapping reliability $\Phi$:

$$\Phi = \sum_{k=1}^{K} \phi_k^e.$$  

(14)

In equation (14), $\phi_k^e$ is mapping reliability corresponding to $G_k^e$ mapping.

Among them,

$$\phi_k^e = \phi_k^e \phi_k^e.$$

(15)

In equation (15), $\phi_k^e$ represents the node reliability mapped from $G_k^e$ to $G_k^e$. $\phi_k^e$ represents the link reliability mapped from $G_k^e$ to $G_k^e$.

In equation (15), $\phi_k^e$ is expressed as

$$\phi_k^e = \prod_{e \in E'_i} \prod_{e \in E'_i} (1 - \alpha_{k,u,i} P^e).$$

(16)

In equation (15), $\phi_k^e$ is expressed as

$$\phi_k^e = \prod_{e \in E'_i} \prod_{e \in E'_i} (1 - \gamma_{k,u,i,j} \beta_{l,u,i} P^e).$$

(17)

Considering the objective functions and constraints of equations (5) and (14), the intelligent mapping problem of secure virtual network can be modeled as an optimization model based on network load minimization and mapping reliability maximization:

$$\begin{align*}
\text{min} & \quad \psi \\
\text{max} & \quad \phi \\
\text{s.t.} & \quad C1 - C12
\end{align*}$$

(18)

In the above formula, $C1 - C12$ represents the risk constraint of secure virtual network under cloud computing.

3.4. Optimization Problem Solving. In the previous section, the optimization problem modeled by equation (18) is a multiobjective optimization problem, which is difficult to be solved by traditional tools. Therefore, this section proposes cloud computing technology to optimize and determine the virtual network mapping strategy. Firstly, the ideal point method is applied to transform the original multiobjective optimization problem into a single objective optimization problem with ideal solutions of two subproblems, namely, the network load minimization subproblem and the mapping reliability maximization subproblem. Secondly, the two-stage virtual network mapping algorithm is applied to solve the two subproblems, respectively, and the ideal solutions of the corresponding subproblems are obtained. Finally, cloud computing technology is used to solve the single objective optimization problem.

The main steps of solving the optimization problem modeled by equation (18) are summarized as follows:

Step 1. Initialize.

Set the particle swarm size $I$, the maximum number of iterations of the algorithm $T_{\text{max}}$, the current number of iterations $t = 0$, $C_{p,i} = \infty$, and $C_{gb} = \infty$ and randomly generate the initial position vector $X_i$ and initial velocity vector $V_i$ of each particle.

Step 2. Fitness function evaluation.

According to the initial position vector $X_i$, firstly, Dijkstra’s algorithm is used to determine the optimal path between $x_{l,u,i}$ and $x_{l,u,i}$, and then calculate the fitness function value $\zeta(X_i)$. If $\zeta(X_i) < \zeta_{p,i}$, set $X_{p,i} = X_i$ and $\zeta_{p,i} = \zeta(X_i)$; if $\zeta(X_i) < \zeta_{gb}$, set $X_{gb} = X_i$ and $\zeta_{gb} = \zeta(X_i)$. 


Step 3. Update speed and position vectors.
Update the velocity vector and position vector of particles, respectively.

Step 4. Terminate judgment conditions.
If $|\zeta_{gb} - \zeta'_{gb}| < \zeta_{th}$, where $\zeta_{th}$ is the decision threshold, the algorithm will terminate; if $t = T_{max}$, the algorithm terminates; otherwise, $t = t + 1$, return to step 2, and continue to execute until the algorithm terminates. The modeling optimization problem is a multiobjective optimization problem. The two-stage virtual network mapping algorithm is applied to solve the two subproblems, respectively; improve the optimal design, optimal plan, optimal management, and optimal control; and obtain the ideal solution of the corresponding subproblem, so as to realize the solution of the optimization problem.

4. Experiment
4.1. Experimental Environment and Parameter Setting. In order to verify the mapping efficiency of this algorithm, simulation experiments are carried out in this paper. In this section, this paper focuses on the simulation environment and parameter setting. Since the network virtualization technology was put forward, the actual prototype and simulation platform based on network virtualization technology have not been fully developed in academia and industry. Therefore, the simulation of the algorithm in this paper is carried out on the self-developed simulation platform. In this simulation, five different underlying physical networks are distributed in a $(200 \times 200)$ two-dimensional plane. These five distributed physical networks are generated by Waxman method model and meet the geographical distribution. Each physical network is connected through inter domain links. The bandwidth of each inter domain link is set as an integer and meets the uniform distribution in the $[300500]$ interval. Each underlying physical network is a medium-sized network in VNE research. See Table 1 for detailed parameters of each physical network.

For the parameter setting of virtual network service, see Table 2 for details. In addition, the simulation experiment will last 100000 time units. A unit of time represents one minute. The simulation time is long enough to obtain the long-term stability of the mapping algorithm. The simulation parameters related to energy consumption are set as follows: $P_{b}$, $P_{max}$, and $P_{l}$ are set to 150 W, 300 W, and 15 W. $P'_{b}$ and $P_{con}$ are set to 150 W and 30 W, respectively. The energy price Pri(t) changes every 3000 unit time. The actual electricity prices of five major cities in China are used to represent the electricity prices of five underlying physical networks. The weight factor is set to 100. $\alpha$ is set to 0.1. $d$ is set to 0.15.

In this section, the cost and power combined optimization algorithm (C-P algorithm), topology segmentation, and clustering analysis algorithm (TS-C algorithm), and TR-CL and EERID-DP algorithm are selected as EERID comparison algorithms. In particular, the C-P algorithm is the most classical heuristic algorithm in the academic world. Ts-c algorithm and TR-CL algorithm are typical energy-saving mapping algorithms in academia. Eri-DP algorithm is a heuristic algorithm based on the direct product value of equation (18). In order to adapt to the simulation conditions of EERID algorithm and meet the requirements (such as energy consumption per hour and node location), all comparison algorithms are modified and updated in this paper.
4.2. Experimental Result. This section consists of two parts. The two parts are as follows: proving that the intelligent mapping algorithm of secure virtual network under cloud computing is a polynomial time mapping algorithm and verifying the mapping efficiency of the intelligent mapping algorithm of secure virtual network under cloud computing.

(1) It is proved that the algorithm in this paper is a polynomial time algorithm: in Table 3, this paper records the average algorithm execution time and mean square deviation time of all selected mapping algorithms. The simulation experiment is run on a small server equipped with Intel Core CPU i7-4790 3.6ghz processor, 32.00 g RAM memory, and Windows 8 operating system. The specific settings of the underlying physical network are consistent with those in the previous section. For virtual network service settings, the number of virtual nodes is fixed at 10. Other parameter settings (such as node connection probability and resource demand) are the same as those described in the previous section. In order to obtain stable and convincing results (i.e., the confidence interval is greater than 95%), this virtual network mapping experiment is repeated 100 times. As can be seen from Table 3, the average mapping execution time of C-P algorithm is 2.76 milliseconds. The average mapping execution time of TS-C algorithm is 3.46 milliseconds. The average mapping execution time of TR-CL algorithm is 2.93 milliseconds. The average mapping execution time of EERID-DP strategy is 3.57 milliseconds. The average mapping execution time of this algorithm is 3.66 milliseconds. As for the mean square deviation results, TR-CL algorithm has the highest mean square deviation among all algorithms. There is little difference in the mean square deviation of other mapping algorithms. Based on the above simulation results, the algorithm can complete the mapping of a given virtual network in polynomial time. Compared with the comparison algorithm selected in this section, although this algorithm needs more
time to calculate the mapping scheme of virtual network, the additional time is not particularly much.

(2) Verify the mapping efficiency of this algorithm: in Figures 5 and 6, this paper plots the average energy consumption (Figure 5) and the average percentage of physical nodes used (Figure 6) of all selected mapping algorithms. By observing the simulation results in Figure 5, we can see that with the continuous extension of simulation time, the energy consumption of all virtual network mapping algorithms increases.

However, in the end, the energy consumption of all mapping algorithms will converge to a stable state, and the occupied physical resources and newly released physical resources will reach a dynamic balance. As can be seen from Figure 5, the energy consumption of the algorithm in this paper is much lower than that of other remaining algorithms, especially compared with the existing energy-saving algorithms (TS-C algorithm and TR-CL). For example, at 60000 unit time points, the energy consumption of the algorithm proposed in this paper is about 8100 RMB/H. Among other mapping algorithms, the best performing TS-C algorithm consumes 9700 RMB/h. The energy consumption of TS-C algorithm is nearly 18% higher than that of this algorithm. There are two reasons for this high energy consumption: the efficient node sorting method of this algorithm and the mapping strategy of integrating the priority mapping of virtual components. As for the node sorting method used in this algorithm, this paper integrates a variety of network topology attributes. The calculated node ranking value can reveal the ability of nodes in the whole network and mapping. For TS-C algorithm, the algorithm only uses the product of CPU and its adjacent link bandwidth as the basis for node ranking. The calculated ranking value cannot fully reflect the mapping ability of nodes in the whole network. As for the mapping strategy adopted by the algorithm, this algorithm adopts the mapping strategy of integrating the priority mapping of used virtual components. Such a strategy is conducive to make full use of the used network elements (nodes) for virtual network mapping, maximize resource utilization, and avoid wasting the use of new network elements.

For TS-C algorithm, TS-C algorithm maps all virtual nodes to physical nodes with high node sorting value. TS-C algorithm does not consider the use of dormant physical nodes as little as possible and only maps the nodes with the highest ranking value. Such a mapping will lead to an increase in the energy cost of TS-C algorithm. At the same time, resource fragmentation and low utilization rate are further aggravated. Figure 5 records a performance comparison of the percentage of physical nodes used by all mapping algorithms. The percentage of nodes used in this algorithm is much lower than other mapping algorithms compared. The conclusions drawn from Figure 6 are consistent with those drawn from Figure 5. The specific reasons and description are similar to the previous paragraph.

5. Conclusion

This paper proposes an intelligent mapping algorithm for secure virtual networks under cloud computing. Network virtualization technology is analyzed, energy consumption model and network load model of virtual network mapping are constructed, mapping reliability is analyzed, virtual network mapping strategy is optimized by cloud computing technology, and virtual network intelligent mapping is realized. The conclusions are as follows:

(1) The percentage of nodes used by the proposed algorithm is much lower than that of other mapping algorithms compared. The energy consumption at 60,000 unit time point is about 8100RMB/h, and the average energy consumption is much lower than that of other methods, indicating that the proposed algorithm has high mapping efficiency.

(2) The proposed algorithm can complete the mapping of a given virtual network in polynomial time, and the percentage of nodes used by the proposed algorithm is far lower than that of other mapping algorithms compared.

(3) The energy consumption of the proposed algorithm is about 8100RMB/h, and the average energy consumption is much lower than that of other methods.

Data Availability

The author can provide all the original data involved in the research.

Conflicts of Interest

The authors indicate that there was no conflict of interest in the study.

References

[1] D. Palanikumar and S. Priya, "Ant colony based graph theory (ACGT) and resource virtual network mapping (RVNM) algorithm for home healthcare system in cloud environment," Multimedia Tools and Applications, vol. 79, no. 5-6, pp. 3743–3760, 2020.

[2] C. Hua, L. Liu, and H. F. Bai, "Physical impairments awareness based virtual network mapping strategy of elastic optical networks," Optoelectronics Letters, vol. 17, no. 1, pp. 36–39, 2021.

[3] J. Zhao and S. Subramaniam, "Virtual network mapping in elastic optical networks with sliceable transponders," Photonic Network Communications, vol. 40, no. 3, pp. 281–292, 2020.

[4] N. Balapuram, J. Raja, and R. Pitchai, "Multicriteria dragonfly graph theory based resource optimized virtual network mapping technique for home medical care service provisioning in cloud," Peer-to-Peer Networking and Applications, vol. 13, no. 6, pp. 1872–1885, 2020.

[5] C. Rong, D. Xie, and C. Qianbin, "SDN virtual network mapping algorithm based on joint optimization of cost and power consumption," Acta Electronica Sinica, vol. 49, no. 8, pp. 1615–1624, 2021.
[6] C. Gang, M. Xiangru, K. Qiaoyan, and Y. Yong, “Virtual software definition network mapping algorithm based on topology segmentation and cluster analysis,” *Computer Applications*, vol. 41, no. 11, pp. 3309–3318, 2021.

[7] A. Miracle and M. Opoku, “Patterned de-duplication on dependable data subcontracting with three error detecting techniques on cloud computing,” *International Journal of Research*, vol. 8, no. 2, pp. 264–266, 2021.

[8] L. Li and H. Liu, “A study on one-dimensional k-coverage problem in wireless sensor networks,” *Wireless Communications and Mobile Computing*, vol. 13, 11 pages, 2013.

[9] M. Park, Y. Lee, I. Yeom, and H. Woo, “Gemma: reinforcement learning-based graph embedding and mapping for virtual network applications,” *IEEE Access*, vol. 9, no. 99, pp. 105463–105476, 2021.

[10] H. Cao, S. Wu, Y. Guo, H. Zhu, and L. Yang, “Mapping strategy for virtual networks in one stage,” *Communications, IET*, vol. 13, no. 14, pp. 2207–2215, 2019.

[11] M. T. Islam, H. Wu, S. Karunasekera, and R. Buyya, “SLA-based scheduling of spark jobs in hybrid cloud computing environments,” *IEEE Transactions on Computers*, vol. 71, no. 5, pp. 1117–1132, 2022.

[12] P. Zhang, X. Pang, Y. Bi, H. Yao, H. Pan, and N. Kumar, “DSCD: delay sensitive cross-domain virtual network embedding algorithm,” *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2913–2925, 2020.

[13] P. Zhang, C. Wang, C. Jiang, and A. Benslimane, “Security-aware virtual network embedding algorithm based on reinforcement learning,” *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 2, pp. 1095–1105, 2021.

[14] D. Thakur and M. Khattu, “Multi-domain virtual network embedding with dynamic flow migration in software-defined networks,” *Journal of Network and Computer Applications*, vol. 162, no. 7, pp. 102639–102639, 2020.

[15] S. Namasudra, “An improved attribute-based encryption technique towards the data security in cloud computing,” *Concurrency and Computation: Practice and Experience*, vol. 31, no. 3, pp. 4364–4364, 2019.

[16] R. A. Haidri, C. P. Katti, and P. C. Saxena, “Cost-effective deadline-aware stochastic scheduling strategy for workflow applications on virtual machines in cloud computing,” *Concurrency & Computation Practice & Experience*, vol. 31, no. 7, pp. 1–24, 2019.

[17] F. Mehmood, I. Ulah, S. Ahmad, and D. H. Kim, “A novel approach towards the design and implementation of virtual network based on controller in future IoT applications,” *Electronics*, vol. 9, no. 4, p. 604, 2020.

[18] S. M. A. Araújo, F. Souza, and G. R. Mateus, “On the analysis of online and periodic virtual network embedding in multi-domain environments,” *International Journal of Networking and Virtual Organisations*, vol. 24, no. 1, pp. 1–21, 2021.

[19] L. Zhuang, S. Tian, M. He, G. Wang, W. Liu, and L. Ma, “Virtual network embedding algorithm via diffusion wavelet,” *IEEE Access*, vol. 7, no. 9, pp. 134145–134157, 2019.

[20] B. Yi, X. Wang, M. Huang, and Y. Zhao, “Novel resource allocation mechanism for SDN-based data center networks,” *Journal of Network and Computer Applications*, vol. 155, no. 4, pp. 102554–102554, 2020.

[21] Z. Li, Z. Lu, S. Deng, and X. Gao, “A self-adaptive virtual network embedding algorithm based on software-defined networks,” *IEEE Transactions on Network and Service Management*, vol. 16, no. 1, pp. 362–373, 2019.

[22] L. Mingtong, “Research on the mechanism and influence factors of urban style building based on cloud computing logistics information,” *Cluster Computing*, vol. 22, no. 6, pp. 13873–13880, 2019.

[23] C. Li, J. Bai, Y. Chen, and Y. Luo, “Resource and replica management strategy for optimizing financial cost and user experience in edge cloud computing system,” *Information Sciences*, vol. 516, no. 2, pp. 1868–1873, 2019.

[24] Y. Wu and S. Jianwen, “Node multi-attribute network mapping algorithm based on topology perception,” *Computer Simulation*, vol. 38, no. 11, pp. 327–354, 2021.

[25] C. Mei, J. Liu, J. Li, L. Zhang, and M. Shao, “5G network slices embedding with sharable virtual network functions,” *Journal of Communications and Networks*, vol. 22, no. 5, pp. 415–427, 2020.

[26] M. Gamal, M. Abohassan, J. Lipman, and W. Ni, “Mapping and scheduling of virtual network functions using multi objective optimization algorithm,” 19th International Symposium on Communications and Information Technologies, vol. 21, no. 11, pp. 25–27, 2019.

[27] J. Wu, S. Zhao, and Y. Peng, “Virtual network mapping for multicast requests over elastic optical networks,” *Optical Engineering*, vol. 58, no. 9, pp. 1–8, 2019.

[28] M. Lu, Y. Gu, and D. Xie, “A dynamic and collaborative multi-layer virtual network embedding algorithm in SDN based on reinforcement learning,” *IEEE Transactions on Network and Service Management*, vol. 17, no. 4, pp. 2305–2317, 2020.

[29] J. Li, W. Shi, H. Wu, S. Zhang, and X. Shen, “Cost-aware dynamic SFC mapping and scheduling in SDN/NFV-enabled space-air-ground integrated networks for internet of vehicles,” *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 1–11, 2021.

[30] P. Zhang, Y. Hong, X. Pang, and C. Jiang, “VNE-HPSo: virtual network embedding algorithm based on hybrid particle swarm optimization,” *IEEE Access*, vol. 8, no. 11, pp. 213389–213400, 2020.

[31] Z. Liu, L. Qin, Q. Liu, J. Liu, and Y. Ding, “A hybrid virtual network mapping algorithm based on threshold load,” *International Journal of Satellite Communications & Networks*, vol. 37, no. 3, pp. 224–233, 2019.

[32] Y. Yue, B. Cheng, X. Liu, M. Wang, B. Li, and J. Chen, “Resource optimization and delay guarantee virtual network function placement for mapping SFC requests in cloud networks,” *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1508–1523, 2021.

[33] H. Cao, H. Zhu, and L. Yang, “Collaborative attributes and resources for single-stage virtual network mapping in network virtualization,” *Journal of Communications and Networks*, vol. 22, no. 1, pp. 61–71, 2020.

[34] C. K. Dehury and P. K. Sahoo, “DVYNE: fitness-based dynamic virtual network embedding in cloud computing,” *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 5, pp. 1029–1045, 2019.

[35] Y. Zhang, J. Qin, X. Li, and S. Zhang, “Improvement of virtual network mapping based on OpenVirteX,” *Journal of Physics: Conference Series*, vol. 1748, no. 3, pp. 32011–32020, 2021.