DE GRUYTER Journal of Intelligent Systems 2025; 34: 20230098

Research Article

Zhiwei Jin*

Cloud computing virtualization technology based on bandwidth resource-aware migration algorithm

https://doi.org/10.1515/jisys-2023-0098
received July 19, 2023; accepted September 13, 2024

Abstract: With the expansion of the Internet, network functions and requirements are becoming more varied. To solve this problem, cloud computing adopts virtualization technology and refers to virtual network function migration to improve network adaptability. However, issues such as resource overload and inefficient migration still exist. Given this, the article proposes a bandwidth-aware virtual network migration algorithm. It combines the risk avoidance simulation learning algorithm with the good and bad distance solution method and applies it to the cloud computing virtual environment. The algorithm monitors bandwidth usage in real-time, identifying potential bottlenecks and considering both resource utilization and bandwidth availability when determining whether to migrate virtual machines. This approach avoids performance degradation due to bandwidth limits. The results showed that the algorithm outperformed others, achieving optimal performance in just 102 iterations and excelling in resource occupancy, utilization, load balancing, migration costs, average migration quantity, latency, and task completion time. This has enabled cloud providers to manage data center network resources in a more efficient manner, thereby enhancing operational efficiency, reducing costs, and delivering more stable and reliable cloud services, which hold significant practical and market value.

Keywords: bandwidth resource perception, VNF migration, rail algorithm, TOPSIS algorithm, cloud computing virtualization

1 Introduction

With the development of the Internet, the Internet of Things technology has also developed, and different fields have gradually developed specialized network requirements. For example, in autonomous driving, extremely low communication delay is required. In manufacturing, it is necessary to minimize packet loss rate and latency as much as possible. In online video interconnection, it is necessary to have the highest possible bandwidth [1,2]. Typically, conventional single-standard networks are only capable of fulfilling one or a few of the desired requirements. Consequently, networks with multiple requirements necessitate the integration of multiple single-standard networks. Considering the network needs of users, integrating and utilizing network resources to achieve intelligent adaptation of the network and providing targeted network services for users is an important research direction in future network development [3]. Upgrading traditional network architectures is difficult to adapt to new network requirements. Cloud computing virtualization technology is proposed, and the concept of network slicing proposed by network operators can better meet current needs [4]. The construction of a network function service chain through the utilization of network functions

*** Corresponding author: Zhiwei Jin**, College of Computer and Artificial Intelligence, Henan Finance University, Zhengzhou, 450046, China, e-mail: jinzhiwei@hafu.edu.cn

virtualization (NFV) technology subsequently facilitates the instantiation of the service chain as a network instance, thereby meeting the corresponding network requirements [5]. Cloud computing technology is a mode of providing computing resources and data storage services through the Internet. Users do not need to own physical hardware to access servers, storage, and applications over the network. The most prevalent application of cloud computing is in machine learning. In this context, cloud computing provides high-performance computing resources and extensive data set storage, enabling enterprises to train intricate machine learning models at a reduced cost. However, cloud computing services mainly rely on the network. In a virtualized environment, the increase in the number of virtual machines may have a negative impact on network bandwidth and latency. The instability of network operation further affects the performance of cloud computing. Concurrently, the implementation of virtualization technology introduces supplementary overhead, including scheduling, resource allocation, and sharing issues at the virtualization layer. These often result in load imbalance. Therefore, this study proposes a virtual network function (VNF) transfer technology based on a bandwidth resource-aware transfer algorithm. The purpose is to further optimize the application effect of cloud computing virtualization technology, achieve more efficient resource management, and promote overall load balancing. The innovation of this research has two points. The first point is to build a virtual network migration model by combining risk aversion simulation learning (RASL) resource awareness and the TOPSIS algorithm. The second point is the introduction of a preventive strategy and a responsive strategy, which reduces the average number of system migrations and reduces migration costs. This study is separated into four parts. The contribution of this research mainly includes two aspects. First, the performance of the cloud computing platform is optimized, and the response speed and concurrency capability of the system are improved. Second, by realizing dynamic resource allocation, the cloud computing platform resource waste and the overall energy consumption are reduced. The first part summarizes and elaborates on relevant research. The second part proposes a bandwidth resource-aware migration algorithm for existing problems. The third part verifies the algorithm proposed in this study. The fourth part is a research summary.

2 Related works

With the boost and maturity of cloud computing virtualization technology, NFV technology has emerged. NFV technology can achieve flexible network deployment and has strong scalability. However, there is a problem of network node resource overload during its application, so researchers have proposed the VNF migration technology. VNF migration technology achieves the virtualization of network functions by deploying traditional physical network resources in the data cloud in a virtual network environment. Yi et al. implemented a new service for business function chains using NFV technology and proposed a VNF migration method, which improved performance and balanced load [6]. Rui et al. proposed a virtual network migration method with reliability as the optimization objective while considering cost. This algorithm had a lower cost and reduced the negative impact caused by resource preemption [7]. Zhang et al. proposed an online delay migration adaptive interference perception algorithm to achieve real-time network changes and VNF migration with excellent reward performance [8]. Li et al. proposed an improved hybrid genetic evolutionary algorithm, which solved the VNF migration problem in dynamic networks, realized more flexible migration, and reduced network latency of different scales [9]. Shang et al. proposed a method to jointly prevent virtual network nodes and routing flows in response to the contradiction between cost and performance in the business function chain. This method ultimately reduced migration costs and prediction errors while maintaining accurate results [10]. Qu et al. studied the scalability of virtual network migration to address the issue of significant network load fluctuations and combined it with the perception of resource overload to reduce training losses [11]. Abdelaal et al. developed an integer linear programming method to simulate the deployment of virtual network migration and proposed a deployment heuristic algorithm considering the redundancy of virtual network migration. This algorithm reduced link utilization and bandwidth consumption [12]. Qu et al. proposed a multi-objective mixed integer optimization scheme to solve the problem of load changing over time in networks. Experiments showed that it reduced Time complexity and guaranteed delay [13].

There are various types of network services nowadays, and it is difficult to update existing services. Not only does it need to consider the issue of construction costs, but there are also compatibility issues between different network devices. Cloud computing virtualization technology is proposed, which allocates computing tasks in cloud resource pools, separates different storage and computing spaces, and completes different software services by combining different requirements. Cerveira et al. analyzed different fault data in virtualized servers and developed a recovery mechanism that can suspend servers due to frequent errors [14]. Rajakumari et al. proposed a fuzzy method based on an updated ant colony optimization algorithm with inertia weights and pheromone trajectories to extend task scheduling in cloud environments, minimize execution time and waiting time, and maximize resource utilization and task scheduling [15]. Eshratifar et al. proposed a self-developed engine to address the latency and energy consumption issues between mobile devices and cloud computing, reducing the load rate and computational load on cloud servers [16]. Badotra and Panda introduced software-defined networks in cloud computing, which increases the manageability of cloud computing as its scalability increases. They explored the importance, advantages, and improvement suggestions of software-defined networks [17]. To make educational activities more convenient and real-time, Trang has constructed a method of using cloud computing technology to achieve virtual classrooms, which has high efficiency and excellent results [18]. Sharma et al. proposed an optimization function to analyze the performance parameters of their developed queuing system. They also proposed two new encryption algorithms for the multi-user optimization problem faced by encrypted data storage in cloud computing. The research emphasized the importance of data from both storage and security perspectives [19]. Yang put forth a topology structure utilizing cluster networks for the analysis of resource utilization in cluster systems. This approach enabled real-time monitoring of resource utilization and load conditions, facilitating improvements in resource utilization and a reduction in system load [20].

The above results show that with the development of communication technology, the traditional network architecture model is no longer suitable for new and diverse needs. NFV technology saves the cost of dedicated network devices and greatly improves the dynamic adaptability of the network by deploying the network to cloud servers and then mapping it to general physical devices. Its shortcomings lie in the high cost of network resources, slow deployment and launch, high static investment, and dynamic operational costs. To solve the above problems, this study combines VNF migration pre-computation and real-time computing and proposes a bandwidth-aware migration algorithm. Then, it introduces the RASL algorithm and the multi-objective decision algorithm TOPSIS for comprehensive optimization, improving migration efficiency and reducing migration costs.

3 Cloud computing virtualization technology in view of bandwidth resource-aware migration algorithm

With the development of the network, traditional single-network architecture has become difficult to meet people's needs [21]. Therefore, cloud computing virtualization technology has been proposed to achieve network adaptability by deploying nodes on virtual networks [22]. When the resources of the virtualization network reach the resource constraint threshold, VNF migration will be triggered [23,24]. The development of migration algorithms has been a prominent area of research to improve the efficiency of migration processes and reduce the occupation of resources during migration. This study proposes a migration algorithm for dynamic bandwidth resource perception from the perspective of bandwidth resource perception.

3.1 Construction of cloud computing virtualization technology in view of bandwidth resource perception migration algorithm

With the widespread application of related technologies, there are already examples of combining software-defined networks (SDN) and NFV to deploy network services onto common network hardware [25]. However, when SDN and NFV technology are mixed, there will be issues of uneven traffic and load. As shown in Figure 1, there are three regions in total, A-1, A-2, and A-3, each with corresponding network management centers. When a large amount of data flows into A-1, the number of connections rapidly increases, but at this time, the network does not make corresponding resource scheduling for this, and other idle resources are not utilized [26]. This leads to an imbalanced load on network nodes. Therefore, the dynamic deployment function of SDN is proposed to adaptively deploy network resources.

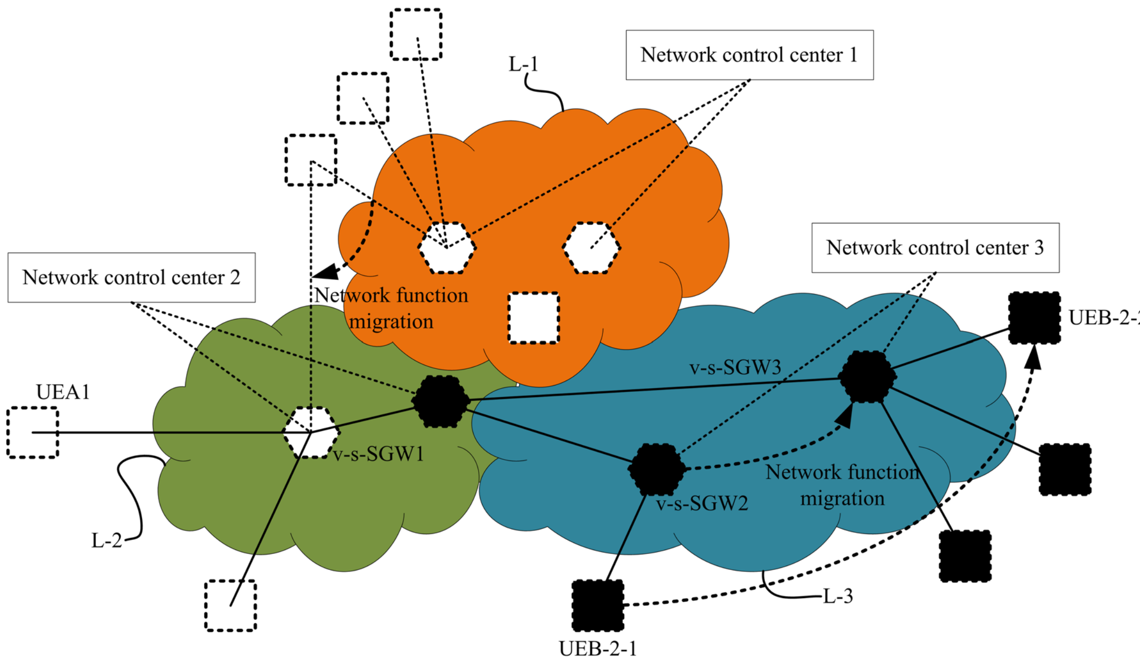

Figure 1: VNF migration diagram. Source: Created by the author.

Therefore, it is necessary to study the network function migration mentioned above. The migration process of VNF can essentially be understood as the redeployment of network architecture, which utilizes a programmable network management mode to dynamically migrate network nodes with resource loads. Dynamic network migration can balance the direct resource allocation in the network control domain. The deployment of VNF networks can be actively migrated according to changes in network business requirements [27]. The SDN control center can perform VNF migration given the location of network users, network QoS, and network QoE requirements, as shown in Figure 2.

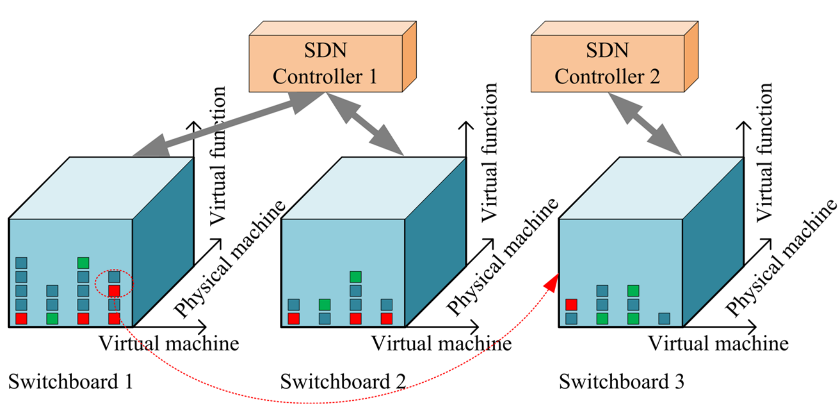

Figure 2: VNF diagram. Source: Created by the author.

The first task is to model the multi-dimensional environment in the network and define the network using SDN and VNF technology, as shown in formula (1). In formula (1), V^{Node} represents the set of nodes. E represents the set of links between nodes

$$G = (V^{\text{Node}}, E). \tag{1}$$

Meanwhile, in view of the resource usage of network nodes, a threshold is set, and the resource constraint threshold of a certain network node is calibrated to T_k, and $T_k \in (0, 1)$, $k \in 1, 2, 3$. The significance of setting a threshold is that if the resources used during network operation exceed the set threshold, migration calculations will be immediately carried out. It should be noted that the threshold T_k of various resources on the same node may vary. If the node resource is set to w_j, then for the node resource, the calculated occupancy rate, storage resource occupancy rate, and network resource occupancy rate are shown in the following equation:

$$\begin{cases} \text{Calculated source usage} = \dfrac{\sum_{i \in V^{\text{VNF}}} n_i^{\text{core}} \cdot a_{ij}}{N_{w_j}^{\text{core}}}, \\ \text{Storage source usage} = \dfrac{\sum_{i \in V^{\text{VNF}}} n_i^{\text{mem}} \cdot a_{ij}}{N_{w_j}^{\text{mem}}}, \\ \text{Network source usage} = \dfrac{\sum_{i \in V^{\text{VNF}}} c_i \cdot a_{ij}}{\sum_{k \neq j, k \in V^{\text{Node}}} C_{j,k}}. \end{cases} \tag{2}$$

In formula (2), N_i^{core} represents the computing resources owned by node i. N_i^{mem} represents the storage resources owned by node i. $C_{j,k}$ represents the link bandwidth between node j and node k, where $j, k \in V^{\text{Node}}$. a_{ij} indicates whether VNF v_i is deployed on node n_j. $a_{ij} = 1$ indicates that v_i is deployed on n_j and $a_{ij} = 0$ indicates that v_i is not deployed on n_j. The purpose of this study is to minimize the overall bandwidth usage in network systems. Further research is conducted on the service function chain (SFC), and the set of SFC is set as follows:

$$F = \{f_1, f_2, \ldots, f_n\}. \tag{3}$$

In formula (3), f_n represents each functional chain, and the minimum bandwidth constraint for each functional chain is ϕ_n. When the VNF waiting for migration belongs to a certain SFC f_n (in this study, only the case where each VNF belongs to one SFC is considered), the sum of its additional bandwidth ϕ_+ and the original bandwidth ϕ_0 should be less than ϕ_n. The set constraint of the SFC is

$$\phi_0 + \phi_+ < \phi_n. \tag{4}$$

If bool is defined, then $b_{ik} = \{0, 1\}$ is used to calibrate VNF v_i and identify whether VNF v_i can be migrated to network node n_k. When variable b_{ik} is 1, it indicates that VNF v_i can migrate to node n_k. On the contrary, it indicates that VNF v_i cannot be migrated to node n_k. The target node n_j for VNF migration must have sufficient idle resources waiting for VNF v_i to migrate in, and it is necessary to ensure that the node's resources do not overload after v_i migrates into n_j. In view of the above, the computational constraints, storage constraints, and bandwidth constraints can be represented by

$$\begin{cases} \text{Computational constraint} = \sum\limits_{l \in V^{\text{VNF}}} n_l^{\text{core}} \cdot a_{lk} + n_i^{\text{core}} \cdot b_{ik} < T_1 \cdot N_{w_k}^{\text{core}}, \\ \text{Storage constraint} = \sum\limits_{l \in V^{\text{VNF}}} n_l^{\text{mem}} \cdot a_{lk} + n_i^{\text{mem}} \cdot b_{ik} < T_2 \cdot N_{w_k}^{\text{mem}}, \\ \text{Bandwidth constraint} = \sum\limits_{l \in V^{\text{VNF}}} c_l \cdot a_{lk} + c_i \cdot b_{ik} < T_3 \cdot \sum\limits_{k \neq j, l \in V^{\text{VNF}}} C_{jk}. \end{cases} \tag{5}$$

In formula (5), n_l^{core} represents the core resources owned by node l. n_l^{mem} represents the storage resources owned by node l. c_l represents the network resources required for VNF l. a_{lk} indicates whether VNF v_l is deployed on node n_k, $a_{lk} = 1$ indicates that v_l is deployed on node n_k, and vice versa, v_l is not deployed on node n_k. b_{ik} is used to identify whether VNF v_i can be migrated to network node n_k. When the variable b_{ik} is 1, it indicates that VNF v_i can be migrated to node n_k. On the contrary, it indicates that VNF v_i cannot be migrated to node n_k. According to the definition of VNF migration, the migration cost function is set as follows:

$$F(\cos t,\ \text{times}) = \lambda \cdot f(\cos t) + \mu \cdot f(\text{times}). \tag{6}$$

In formula (6), $f(\cos t)$ represents the migration cost. $f(\text{times})$ represents the number of transfers, where λ and μ represent adjustable weight coefficients. $f(\cos t)$ includes the total computational time for migration and the time required to implement the hardware and software migration process. $f(\text{times})$ represents the number of times migration occurred during the VNF v_i migration process. By migrating time $T(v_i, n_j, n_k)$ and migrating storage resources n_{vi}^{mem}, the migration bandwidth $B(v_i)$ of VNF node v_i from n_j to node n_k can be obtained from the migration bandwidth, migration time, and migration traffic, as shown in the following equation:

$$B(v_i) = \frac{n_{vi}^{\text{mem}}}{T(v_i, n_j, n_k)}. \tag{7}$$

In formula (7), n_{vi}^{mem} represents the storage resources occupied by VNF v_i. $T(v_i, n_j, n_k)$ represents the time taken for VNF v_i migration. SFC is a VNF node chain that is constructed in view of actual needs and is specifically sorted. After VNF instantiation, it can generate actual available network links. In the process of VNF migration, replacing it with a new SFC based on the actual needs of the network can reduce the cost of network migration. The deployment and composition diagram of the SFC is shown in Figure 3.

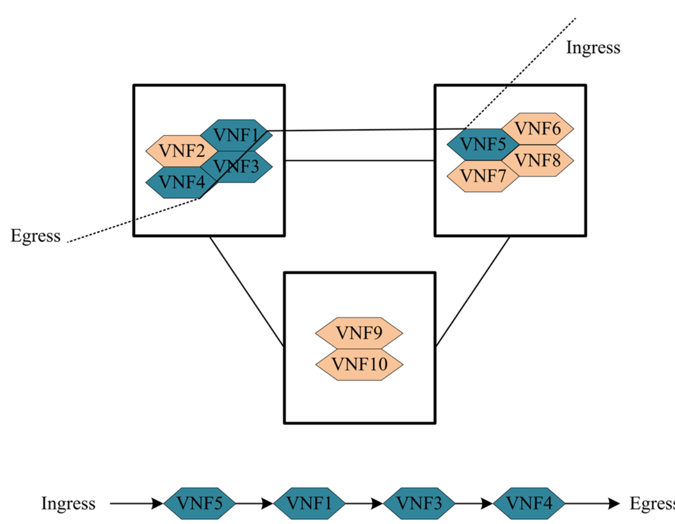

Figure 3: SFC deployment and composition. Source: Created by the author.

3.2 Optimization of cloud computing virtualization technology in view of bandwidth resource perception migration algorithm

During the migration process of VNF, there may be situations where the bandwidth is extremely high [28,29]. As shown in Figure 4, when the resources of node A exceed the threshold constraint limit, VNF migration is required. When the branch VNF1 of VNF needs to be migrated to node C, this migration will also cause SFC to migrate, further increasing bandwidth. The virtual infrastructure management network controller will reassign paths to SFC, and at this point, the additional bandwidth of the current link is 3.

At this time, there may be a situation of network traffic forwarding in the opposite direction on the link, and the network bandwidth will double at this time. Figure 5 shows the VNF migration situation. It is typical for multiple SFC chains to be necessary for the implementation of a VNF instance. This presents a scenario that warrants further examination. In a VNF migration network, a single VNF may be utilized by multiple SFCs, and

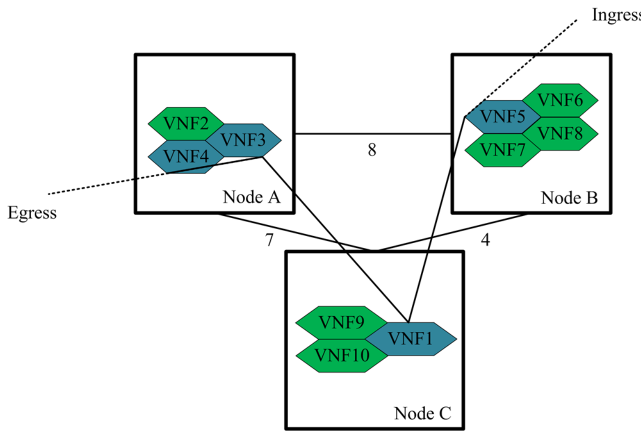

Figure 4: VNF migration. Source: Created by the author.

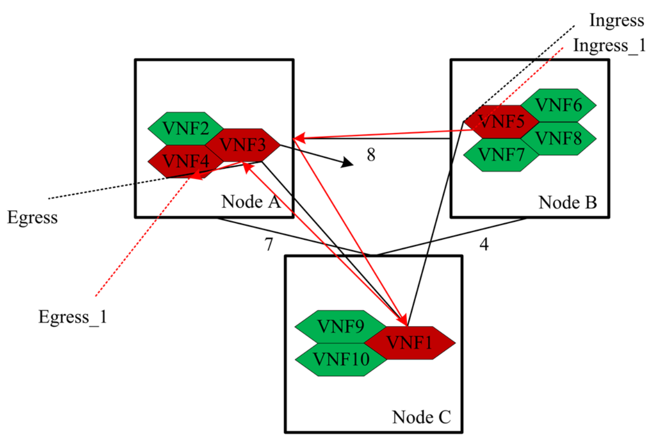

Figure 5: VNF migration overload. Source: Created by the author.

the bandwidth of these SFCs may vary. However, they all adhere to a minimum constraint, and the aggregate additional bandwidth of the VNF migration and the original bandwidth must remain below the minimum bandwidth constraint of the SFC to which this VNF belongs. This study will further investigate and optimize the bandwidth constraints of a single SFC in VNF migration.

To reduce the migration cost of VNF and further enhance the dynamic flexibility of virtual network migration, this study proposes a bandwidth resource awareness method in view of pre-computing and real-time computing. This method combines this perception method for VNF transfer. This method can effectively reduce the computational cost of VNF migration and further reduce the number of network system migrations and the time cost during migration. The implementation process of the migration algorithm in view of the cooperation of migration pre-computing and real-time computing is shown in Figure 6.

Various resources on each node have corresponding thresholds. When the occupancy of a certain resource in the node exceeds the threshold, VNF migration is performed. It uses dynamic perception of the resource usage of each node to determine whether VNF migration is necessary. Therefore, the VNF migration algorithm is introduced into the RASL dynamic resource awareness algorithm. When the load of a certain resource is uneven, VNF migration is carried out in view of the type of resource that exceeds the limit. Nodes with more load can be migrated to destination nodes with idle resources. The calculation of network occupancy, storage occupancy, and network occupancy is shown in the following equation:

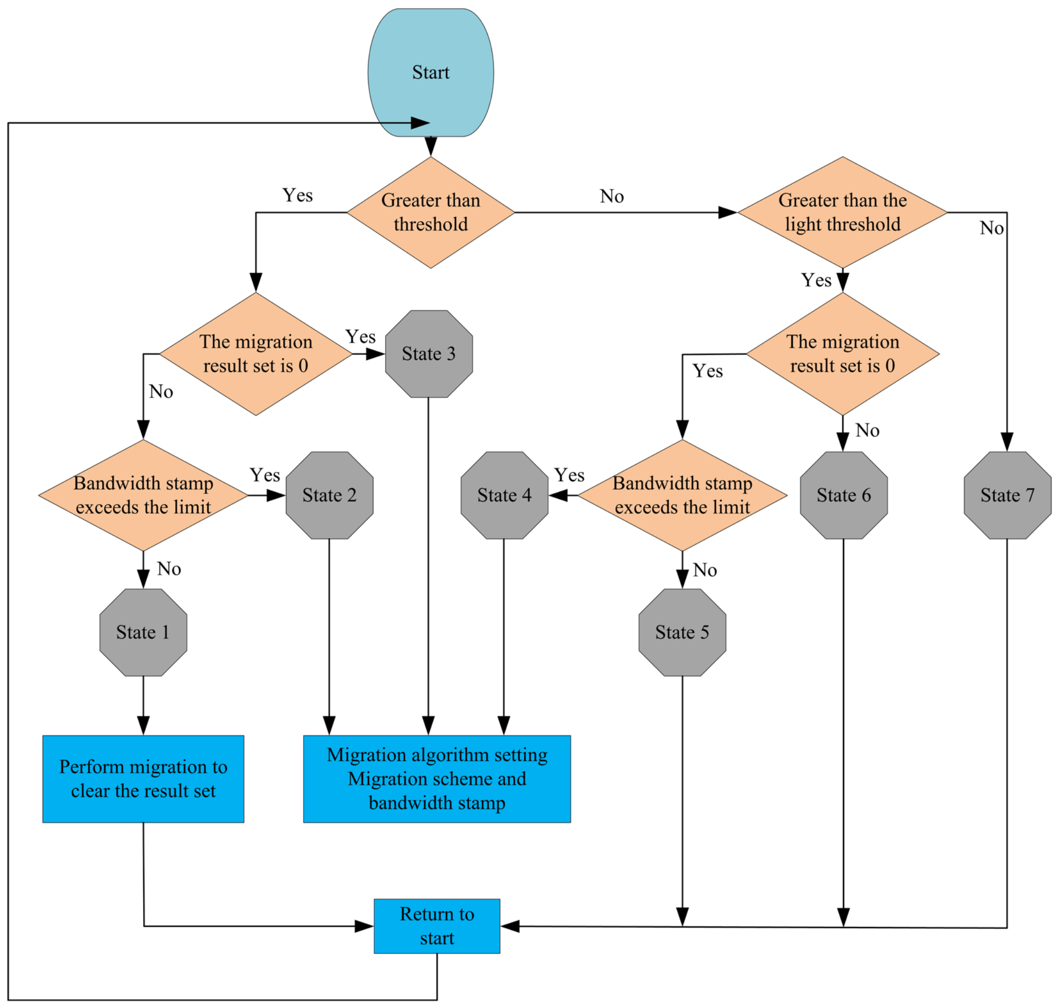

Figure 6: Migration of predictive and real-time computing synergistic processes. Source: Created by the author.

$$\begin{cases} \text{Calculated source usage} = \dfrac{n_i^{\text{core}}}{N_{w_j}^{\text{core}}}, \\ \text{Storage source usage} = \dfrac{n_i^{\text{mem}}}{N_{w_j}^{\text{mem}}}, \\ \text{Network source usage} = \dfrac{c_i}{\sum_{k \neq j, k \in V^{\text{Node}}} C_{j,k}}. \end{cases} \tag{8}$$

By utilizing bandwidth resource perception and setting the dynamic weight of RASL, the migration coefficient θ_i is obtained as shown in formula (9). θ_i can provide intuitive feedback on the bandwidth resource usage of nodes. When $\theta_i > 1$ is used, it indicates that the node resource usage is unbalanced. When $\theta_i < 1$, it indicates that the resource occupancy situation is stable. θ_i can display whether the overload situation of the node is severe, and the larger the θ_i, the more severe the overload degree

$$\theta_i = \sum_k \left[a_k \frac{1}{1 - x_k} + (1 - a_k)(1 - x_k) \right] x_{ik}, \quad K = (\text{Core}, \ \text{Mem}, \ \text{Com}). \tag{9}$$

In formula (9), a_k represents the overload of Class k resources of the node. x_k represents the proportion of k resources to the total resources of the node. x_{ik} represents the occupancy rate of the k class resources of the node by VNF migration v_i. According to the migration index θ_i, resources with a higher migration index can be selected for high-priority migration. According to the end-to-end performance constraint requirements of the VNF to be migrated to the SFC where it is located, formula (10) is obtained.

$$a_{ij} \cdot b_{ik} \cdot \Delta\phi_{jk+} + \phi_0 < \phi_{\max}, v_i \in f_n. \tag{10}$$

According to the constraint conditions of formula (10), the migration is carried out by selecting nodes with large remaining resources to ensure that resource overloading does not occur after the migration. It then uses the TOPSIS algorithm to balance the migration index during the process of selecting nodes for VNF migration. According to the definition of TOPSIS, the first is to calculate the positive ideal solution best-Node, as shown in the following equation:

$$T_{i+} = \min\{T_{ki}|k \in N_3\}, \quad i \in \{1, 2, 3\}. \tag{11}$$

According to formula (11), it calculates the minimum utilization rate of each resource in the selected node as the occupancy rate (T_{1+}, T_{2+}, T_{3+}) of the ideal solution, and obtains the maximum additional bandwidth of all selected nodes as ϕ_+, as shown in formula (12). The additional bandwidth generated by VNF migration is taken as a positive ideal solution.

$$\phi_+ = \min\{\tau_k|k \in N_3\}. \tag{12}$$

After calculating the positive ideal solution and using it to obtain the maximum additional bandwidth ϕ_+, according to the TOPSIS algorithm, the negative ideal solution worst-Node is calculated, as shown in the following equation:

$$T_{i-} = \max\{T_{ki}|k \in N_3\}, \quad i \in \{1, 2, 3\}. \tag{13}$$

According to formula (13), combined with the RASL dynamic resource awareness algorithm for analysis, the maximum resource utilization rate on the VNF node to be migrated can be obtained. The resource utilization rate is taken as the occupancy rate (T_{1-}, T_{2-}, T_{3-}) of the negative ideal solution. After the RASL analysis, the worst additional bandwidth ϕ_- of the negative ideal solution can be obtained, as shown in the following equation:

$$\phi_- = \max\{\phi_k|k \in N_3\}. \tag{14}$$

Finally, the Euclidean distance θ_{ok+} from the node w_k to the positive ideal solution and the Euclidean distance w_k from the node w_k to the negative ideal solution were calculated. The approximate distance θ_{ok} from the node to the positive ideal solution is calculated, as shown in the following equation:

$$\begin{cases} \theta_{ok+} = \sqrt{\sum_{i=1}^{3}[\gamma_i(T_{ki} - T_{i+})]^2 + [\gamma_4(\phi_k - \phi_+)]^2}, \\ \theta_{ok-} = \sqrt{\sum_{i=1}^{3}[\gamma_i(T_{ki} - T_{i-})]^2 + [\gamma_4(\phi_k - \phi_-)]^2}, \\ \theta_{ok} = \dfrac{\theta_{ok-}}{\theta_{ok+} + \theta_{ok-}}. \end{cases} \tag{15}$$

In formula (15), γ_i represents the preset defined weights of each indicator. θ_{ok} represents the proximity distance between the node and the positive and negative ideal solution. The larger the θ_{ok}, the closer the node is to its corresponding ideal solution. Therefore, in the migration process of VNF, the node with the largest θ_{ok} should be selected as the migration destination node. When the demand changes in the network, the migration of VNF is triggered. At this time, the destination node of the migration will evaluate the resource situation based on the calculated weight. Only when the resource demand is met will other resources be subjected to further consideration. By setting weights, the optimal candidate migration nodes can be provided for VNF migration, avoiding ineffective migration and improving network migration efficiency.

4 Evaluation of cloud computing virtualization technology in view of bandwidth resource-aware migration algorithms

To verify the effectiveness of the bandwidth resource-aware migration algorithm proposed in this study, the network of an ISP is taken as an example. The initial network nodes in the simulated network are set to be 100, each of which contains computer network resources, such as storage, network, and arithmetic. About 500 network links are generated based on these nodes. It uses a fixed weight algorithm, simple instant algorithm, and PRT algorithm to compare node migration performance with the RT algorithm (RASL&TOPSIS) proposed in this study. The comparison and expected benefits of several algorithms are shown in Table 1.

Table 1: Comparison and characteristics of algorithms

Algorithm	The similarity with the RT algorithm	The difference with the RT algorithm	Characteristic
RT algorithm (RASL&TOPSIS)	—	—	To obtain the optimal migration scheme of the optimization target, but the complexity is high
Fixed weight algorithm (FW) [30]	TOPSIS algorithm was adopted.	Traditional waiting for migration, no resource awareness	The computational complexity is reduced, and the computation cannot be accelerated
Simple real-time algorithm (SR) [31]	RASL resource awareness was used	Nodes are selected with performance constraints	This reduces the complexity and system stability
PRT algorithm (PRT) [32]	TOPSIS and RASL algorithms are adopted	A pre-calculation mechanism is used	Greatly reduce the migration calculation overhead

First, it compares the convergence of the four algorithms, and the comparison results are shown in Figure 7. Figure 7 shows that the improved algorithm in this study can achieve the optimal target accuracy and loss value after training 102 times, which is 38 times less than the FW algorithm, 57 times less than the SR algorithm, and 19 times less than the PRT algorithm. This indicates that the convergence of the improved bandwidth resource-aware migration algorithm is superior to other algorithms.

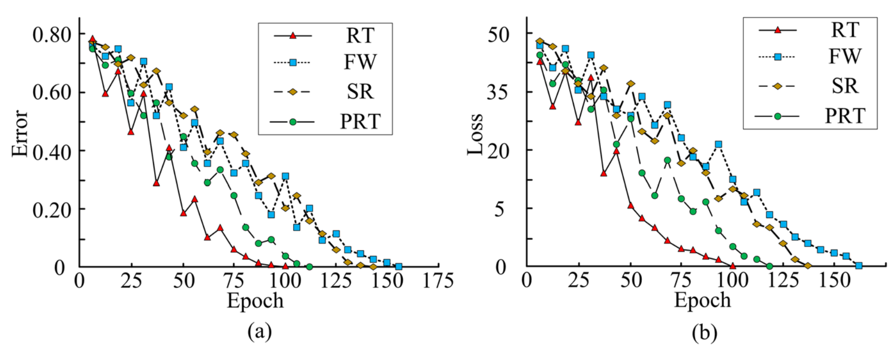

Figure 7: Convergence of four algorithms: (a) error and (b) loss. Source: Created by the author.

Then, it compares the resource utilization and load balancing of the four algorithms. The results are shown in Figure 8. Figure 8 shows that with the increase of migration nodes, the improved bandwidth resource-aware migration algorithm has increased its resource utilization rate from 54 to 71%, leading by 7–17% relative to other algorithms. The improved algorithm in this study performs well in terms of load balancing. As the number of migrated nodes increases, its load balancing degree increases from 65 to 86%, leading by 6–19% relative to other algorithms. This indicates that the improved bandwidth resource-aware migration algorithm in this study performs well in terms of resource utilization and load balancing when used for node migration.

It compares the average migration times and average migration costs of the four algorithms, and the results are shown in Figure 9. Figure 9 shows that the number of network function migrations and migration

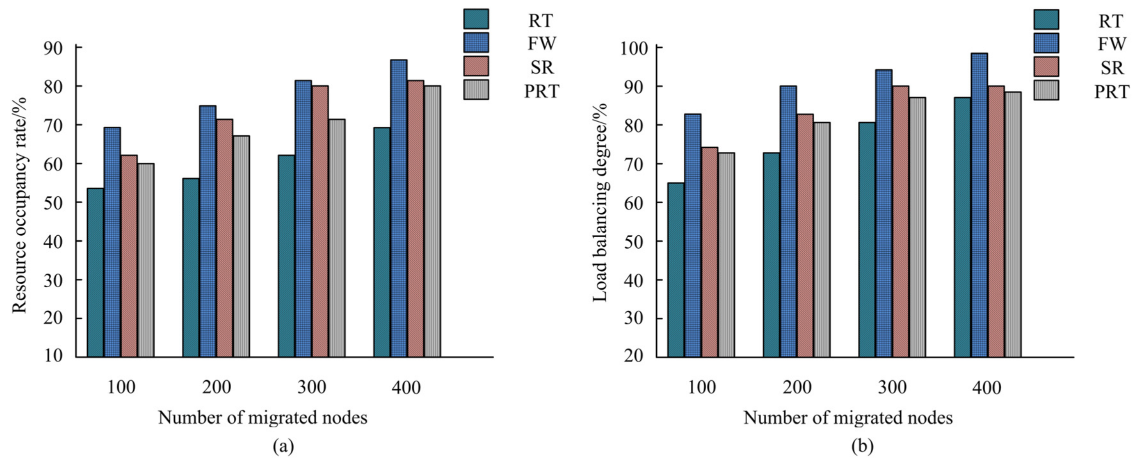

Figure 8: Resource usage and load balancing of four algorithms: (a) resource occupancy rate and (b) load balancing degree. Source: Created by the author.

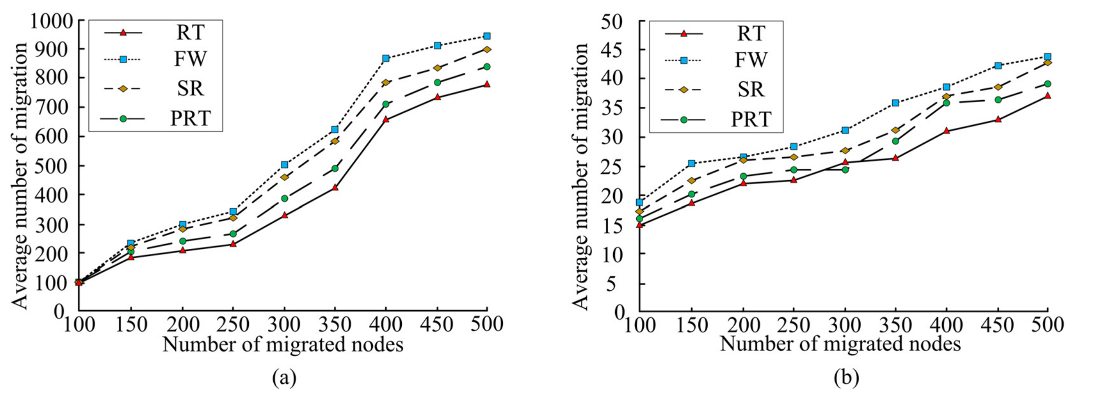

Figure 9: Average number and cost of migration of four algorithms: (a) average number of migration and (b) average cost of migration. Source: Created by the author.

costs per unit system time increase with the increase of the number of network nodes. Due to the use of collaborative resource awareness and multi-objective decision-making in the RT algorithm, the migration decision of the RT algorithm is more comprehensive. Due to the introduction of the TOPSIS algorithm, every migration of the RT algorithm is the most effective, resulting in the best average migration quantity and migration cost. Compared with the FW algorithm, SR algorithm, and PRT algorithm, the average migration quantity has decreased by 17–28%, 12–19%, and 4–11%, respectively. Relative to the FW algorithm, SR algorithm, and PRT algorithm, the migration cost has decreased by 12–21%, 8–15%, and 5–9%, respectively.

Then, it compares the energy consumption and latency of the four algorithms, and the results are shown in Figure 10. Figure 10 shows that as the number of migrated nodes increases, their energy consumption and latency also increase, but the overall performance of the RT algorithm is better. It indicates that the energy consumption of the RT algorithm is 21–32%, 14–21%, and 9–12% less than that of the FW algorithm, SR algorithm, and PRT algorithm, respectively. Its delay is shortened by 23–40%, 19–35%, and 15–21% relative to the FW algorithm, SR algorithm, and PRT algorithm, respectively. Therefore, in terms of energy consumption and latency, the RT algorithm proposed in this study can be more effectively applied in cloud computing virtualization.

To demonstrate that the improved algorithm has better migration performance, the Cloudsim simulation tool is used to set random tasks. Furthermore, the resource utilization and task execution time of four algorithms are compared as a function of the number of tasks. The results are shown in Figure 11. Figure 11 shows that as the number of tasks increases, the task execution time increases and resource utilization decreases. The improved

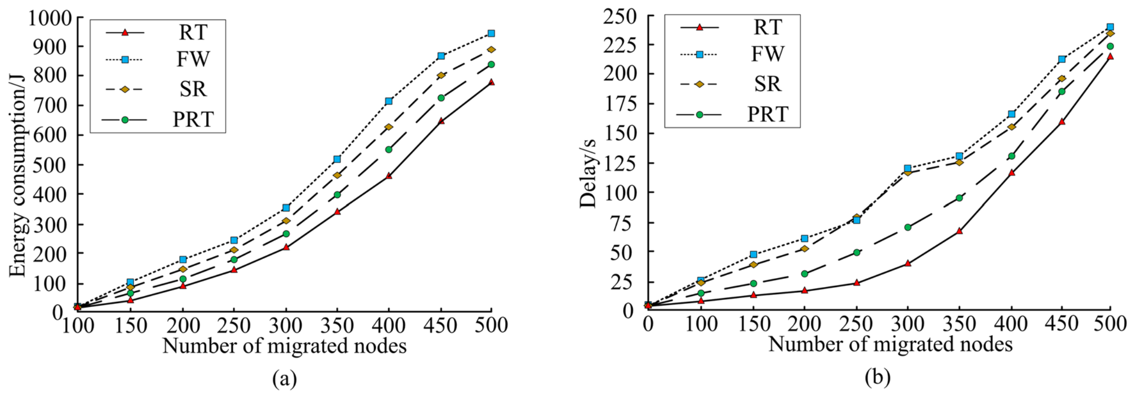

Figure 10: Energy consumption and delay of four algorithms: (a) energy consumption and (b) delay. Source: Created by the author.

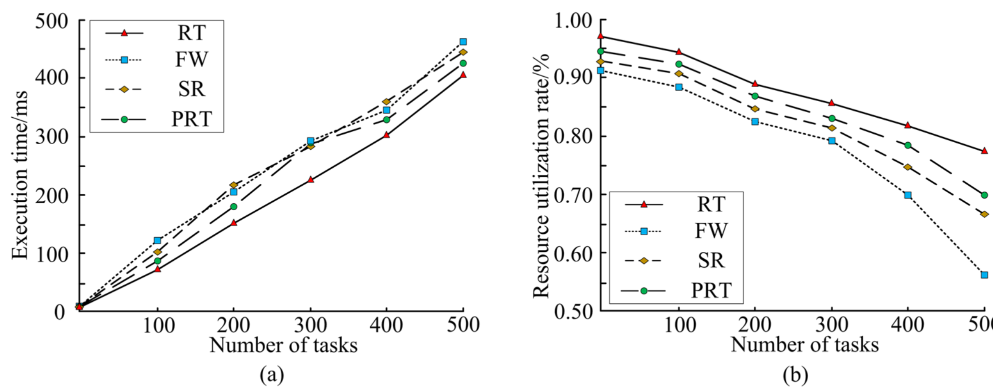

Figure 11: Energy consumption and delay of four algorithms: (a) execution time and resource utilization rate. Source: Created by the author.

algorithm in this study has reduced task execution time by 4–8%, 3–7%, and 2–4% relative to the FW algorithm, SR algorithm, and PRT algorithm, respectively. Relative to the FW algorithm, SR algorithm, and PRT algorithm, the resource utilization rate has increased by 7–38%, 5–25%, and 2–18%, respectively.

Finally, the total migration resource cost and migration completion time of the four algorithms are compared, and the results are shown in Figure 12. Figure 12 shows that as the number of migration nodes increases, the migration completion time also increases. The improved algorithm in this study has shortened the migration

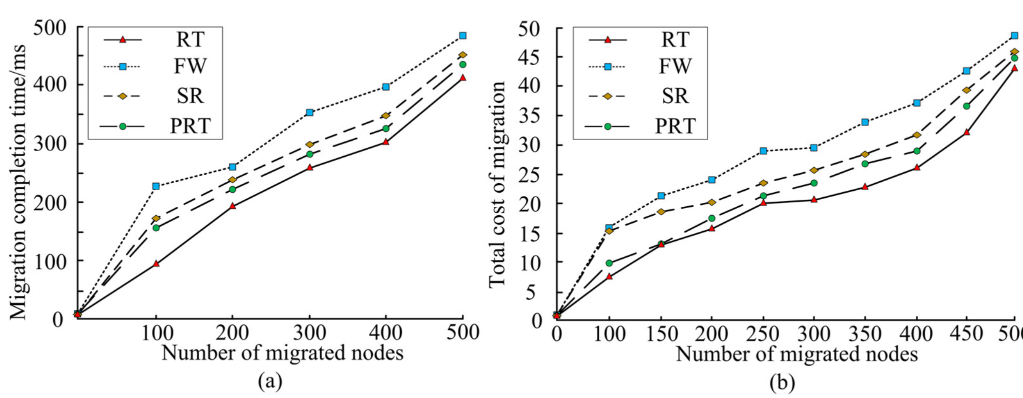

Figure 12: Migration completion time and total overhead of four algorithms: (a) migration completion time and (b) total cost of migration. Source: Created by the author.

completion time by 23–56%, 19–27%, and 11–15% relative to the FW algorithm, SR algorithm, and PRT algorithm, respectively. In terms of overall resource expenditure, the improved algorithm in this study has reduced by 24–38%, 11–21%, and 5–9% relative to the FW algorithm, SR algorithm, and PRT algorithm, respectively. In summary, the comparison with the FW algorithm, SR algorithm, and PRT algorithm fully proves that the improved algorithm in this study performs well and can be effectively applied in cloud computing and virtualization.

5 Conclusion

In the context of the increasingly diverse functions and requirements of the modern Internet, VNF migration technology has emerged to enhance the adaptive capabilities of cloud computing environments. However, the challenges of resource overload and low migration efficiency underscore the limitations of contemporary cloud virtualization technologies. Given this, an algorithm was introduced that amalgamates RASL with a superior–inferior solution distance method dedicated to improving the efficiency of cloud computing virtualization. This algorithm aimed to accurately locate bandwidth bottlenecks by continuously monitoring the network status within cloud data centers. It also combined the virtual machine migration decision-making process that considers the utilization of target host resources and network bandwidth availability. This ensured that performance did not degrade due to bandwidth constraints. Analysis of the data revealed that in terms of migration completion time, the proposed algorithm achieved a reduction of 23–56%, 19–27%, and 11–15% compared to the FW, SR, and PRT algorithms, respectively. Furthermore, it reduced the total resource expenditure by 24–38%, 11–21%, and 5–9% against the same benchmarks. The algorithm significantly enhanced the management efficiency of network resources within data centers for cloud service providers, optimizing the virtualization environment while reducing operational costs and improving the stability and reliability of cloud services. However, the proposed method still has some limitations; that is, it reduces the migration time to a certain extent, resulting in the increase of additional migration actions, which makes the server face greater pressure. Therefore, a heuristic search algorithm should be used in the future to reduce the migration action of the split to improve the efficiency of the algorithm. At the same time, the generalization of the algorithm is further enhanced, and the migration decision is dynamically adjusted through a more powerful mechanism, thereby improving the performance of the cloud computing platform.

Funding information: The author states no funding involved.

Authors contribution: Zhiwei Jin confirm the sole responsibility for the conception of the study, presented results and manuscript preparation.

Conflict of interest: The author states no conflict of interest.

Data availability statement: All data generated or analyzed during this study are included in this published article.

References

[1] Cheng X, Fan Y. Research and design of intelligent speech equipment in smart English language lab based on Internet of Things technology. Procedia Comput Sci. 2022;198(1):505–11. doi: 10.1016/j.procs.2021.12.277.
[2] Nappi I, de Campos Ribeiro G. Internet of Things technology applications in the workplace environment: A critical review. J Corp Real Estate. 2020;22(1):71–90. doi: 10.1108/JCRE-06-2019-0028.
[3] Rao PM, Deebak BD. Security and privacy issues in smart cities/industries: technologies, applications, and challenges. J Amb Intel Hum Comp. 2023;14(8):10517–53. doi: 10.1007/s12652-022-03707-1.
[4] Jawed MS, Sajid M. A comprehensive survey on cloud computing: architecture, tools, technologies, and open issues. Int J Cloud Appl Comput. 2022;12(1):1–33. doi: 10.4018/IJCAC.308277.

[5] Mai L, Ding Y, Zhang X, Fan L, Yu S, Xu Z. Energy efficiency with service availability guarantee for network function virtualization. Future Gener Comput Syst. 2021;119(1):140–53. doi: 10.1016/j.future.2021.02.002.
[6] Yi B, Wang X, Huang M, Li K. Design and implementation of network-aware VNF migration mechanism. IEEE Access. 2020;8(1):44346–58. doi: 10.1109/ACCESS.2020.2978002.
[7] Rui L, Chen X, Gao Z, Li W, Qiu X, Meng L. Petri net-based reliability assessment and migration optimization strategy of SFC. IEEE Trans Netw Serv Manag. 2020;18(1):167–81. doi: 10.1109/TNSM.2020.3045705.
[8] Zhang Q, Liu F, Zeng C. Online adaptive interference-aware VNF deployment and migration for 5G network slice. IEEE/ACM Trans Netw. 2021;29(5):2115–28. doi: 10.1109/TNET.2021.3080197.
[9] Li B, Cheng B, Liu X, Wang M, Yue L, Chen J. Joint resource optimization and delay-aware virtual network function migration in data center networks. IEEE Trans Netw Serv Manag. 2021;18(3):2960–74. doi: 10.1109/TNSM.2021.3067883.
[10] Shang X, Liu Z, Yang Y. Online service function chain placement for cost-effectiveness and network congestion control. IEEE Trans Comput. 2020;71(1):27–39. doi: 10.1109/TC.2020.3035991.
[11] Qu K, Zhuang W, Shen X, Li X, Rao J. Dynamic resource scaling for VNF over nonstationary traffic: A learning approach. IEEE Trans Cognit Commun Netw. 2020;7(2):648–62. doi: 10.1109/TCCN.2020.3018157.
[12] Abdelaal MA, Ebrahim GA, Anis WR. High availability deployment of virtual network function forwarding graph in cloud computing environments. IEEE Access. 2021;9(1):53861–84. doi: 10.1109/ACCESS.2021.3068342.
[13] Qu K, Zhuang W, Ye Q, Shen X, Rao J. Dynamic flow migration for embedded services in SDN/NFV-Enabled 5G core networks. IEEE Trans Commun. 2020;68(4):2394–408. doi: 10.1109/TCOMM.2020.2968907.
[14] Cerveira F, Barbosa R, Madeira H, Araujo F. The effects of soft errors and mitigation strategies for virtualization servers. IEEE Trans Cloud Comput. 2020;10(2):1065–81. doi: 10.1109/TCC.2020.2973146.
[15] Rajakumari K, Kumar MV, Verma G, Balu S, Sharma DK, Sengan S. Fuzzy based ant colony optimization scheduling in cloud computing. Comput Syst Sci Eng. 2022;40(2):581–92. doi: 10.32604/csse.2022.019175.
[16] Eshratifar AE, Abrishami MS, Pedram M. Joint DNN: An efficient training and inference engine for intelligent mobile cloud computing services. IEEE Trans Mob Comput. 2021;20(2):565–76. doi: 10.1109/TMC.2019.2947893.
[17] Badotra S, Panda SN. Software defined networking: a crucial approach for cloud computing adoption. Int J Cloud Comput. 2022;11(2):123–37. doi: 10.1504/IJCC.2022.122028.
[18] Trang TTT. From practice to prediction about the trend to develop virtual classroom model on cloud computing. VNUHCM J Soc Sci Humanit. 2021;5(2):1019–25. doi: 10.32508/stdjssh.v5i2.664.
[19] Sharma D, Kumar G, Sharma R. Analysis of heterogeneous data storage and access control management for cloud computing under M/M/c queueing model. Int J Cloud Appl Comput (IJCAC). 2021;11(3):58–71. doi: 10.4018/IJCAC.2021070104.
[20] Yang F. Cloud computing virtual resource dynamic system allocation and application based on system architecture. Dyn Syst Appl. 2021;30(5):753–70. doi: 10.46719/dsa20213056.
[21] Wang T, Xu H, Liu F. Multi-resource load balancing for virtual network functions. 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS). IEEE; 2017. p. 1322–32. doi: 10.1109/ICDCS.2017.233.
[22] Xu F, Liu F, Xi H. Heterogeneity and interference-aware virtual machine provisioning for predictable performance in the cloud. IEEE Trans Comput. 2015;65(8):2470–83. doi: 10.1109/TC.2015.2481403.
[23] Xu F, Liu F, Liu L, Xi H, Li B, Li B. iAware: Making live migration of virtual machines interference-aware in the cloud. IEEE Trans Comput. 2013;63(12):3012–25. doi: 10.1109/TC.2013.185.
[24] Fei X, Liu F, Xu H, Jin H. Adaptive VNF scaling and flow routing with proactive demand prediction. IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE; 2018. p. 486–94. doi: 10.1109/INFOCOM.2018.8486320.
[25] Zeng C, Liu F, Chen S, Jiang W, Li M. Demystifying the performance interference of co-located virtual network functions. IEEE INFOCOM 2018-IEEE Conference on Computer Communications. IEEE; 2018. p. 765–73. doi: 10.1109/INFOCOM.2018.8486246.
[26] Xiao Y, Zhang Q, Liu F, Wang J, Zhao M, Zhang Z, Zhang J. NFVdeep: Adaptive online service function chain deployment with deep reinforcement learning. Proceedings of the International Symposium on Quality of Service; 2019. p. 1–10. doi: 10.1145/3326285.3329056.
[27] Fei X, Liu F, Xi H, Li B. FlexNFV: Flexible network service chaining with dynamic scaling. IEEE Netw. 2020;34(4):203–9. doi: 10.1109/MNET.001.1900483.
[28] Xu F, Liu F, Xi H, Vasilakos AV. Managing performance overhead of virtual machines in cloud computing: A survey, state of the art, and future directions. Proc IEEE. 2013;102(1):11–31. doi: 10.1109/JPROC.2013.2287711.
[29] Sarkar A, Biswas A, Kundu M. Development of Q-rung Orthopair trapezoidal fuzzy Einstein aggregation operators and their application in MCGDM problems. J Comput Cognit Eng. 2022;1(3):109–21. doi: 10.47852/bonviewJCCE2202162.
[30] Sun J, Liu F, Wang H, Ahmed M, Li Y, Liu M. Efficient VNF placement for Poisson arrived traffic. IEEE Trans Netw Serv Manag. 2021;18(4):4277–93. doi: 10.1109/TNSM.2021.3102583.
[31] Akbari M, Abedi MR, Joda R, Pourghasemian M, Mokari N, Erol-Kantarci M. Age of information aware VNF scheduling in industrial IoT using deep reinforcement learning. IEEE J Sel Areas Commun. 2021;39(8):2487–500. doi: 10.1109/JSAC.2021.3087264.
[32] Shi Y, Wang P, Zhu X, Zhu H. Reconfigurable digital satellite-borne base station design and virtual function fast migration algorithm. Sensors. 2023;23(17):7591–3. doi: 10.3390/s23177591.

Appendix

All variable symbols

Variable symbols	Meanings
V^{Node}	Set of nodes
E	Set of links between nodes
T_k	Threshold and calibrate the resource constraint threshold of a certain network node
w_j	Node resource
N_i^{core}	Computing resources owned by node i
N_i^{mem}	Storage resources owned by node i
$C_{j,k}$	Link bandwidth between node j and node k
a_{ij}	Whether VNF v_i is deployed on node n_j
f_n	Each functional chain
ϕ_+	Sum of its additional bandwidth
ϕ_0	Original bandwidth
ϕ_n	Minimum bandwidth constraint for each functional chain
n_l^{core}	Core resources owned by node l
n_l^{mem}	Storage resources owned by node l
c_l	Network resources required for VNF l
a_{lk}	Whether VNF v_l is deployed on node n_k
b_{ik}	Using to identify whether VNF v_i can be migrated to network node n_k
$f(\cos\ t)$	Migration cost
$f(\mathrm{times})$	Number of transfers
λ	Adjustable weight coefficients
μ	Adjustable weight coefficients
$B(v_i)$	The migration bandwidth of VNF node v_i from n_j to node n_k
n_{vi}^{mem}	Storage resources occupied by VNF v_i
$T(v_i, n_j, n_k)$	The time taken for VNF v_i
θ_i	Migration coefficient
a_k	The overload of Class k resources of the node
x_k	The proportion of k resources to the total resources of the node
x_{ik}	The occupancy rate of the k class resources of the node by VNF migration v_i
ϕ_-	The worst additional bandwidth
θ_{ok+}	The Euclidean distance between node w_k and the positive ideal solution
θ_{ok-}	The Euclidean distance w_k between node w_k and the negative ideal solution
θ_{ok}	The approaching distance between node and the positive ideal solution
γ_i	The preset defined weights of each indicator