Accepted Manuscript

Mobile-aware service function chain migration in cloud-fog computing

Dongcheng Zhao, Dan Liao, Gang Sun, Shizhong Xu, Victor Chang

PII: S0167-739X(18)32532-9
DOI: https://doi.org/10.1016/j.future.2019.02.031
Reference: FUTURE 4784

To appear in: *Future Generation Computer Systems*

Received date : 17 October 2018
Revised date : 16 February 2019
Accepted date : 18 February 2019

Please cite this article as: D. Zhao, D. Liao, G. Sun et al., Mobile-aware service function chain migration in cloud-fog computing, *Future Generation Computer Systems* (2019), https://doi.org/10.1016/j.future.2019.02.031

Mobile-aware Service Function Chain Migration in Cloud-Fog Computing

Dongcheng Zhao[a,b], Dan Liao[a,c,*], Gang Sun[a], Shizhong Xu[a], and Victor Chang[d]

[a]*Key Lab of Optical Fiber Sensing and Communications (Ministry of Education), University of Electronic Science and Technology of China, China*

[b]*Science and Technology on Communication Networks Laboratory*

[c]*Chengdu Research Institute, University of Electronic Science and Technology of China, China*

[d]*Xi'an Jiaotong Liverpool University, Suzhou, China*

HIGHLIGHTS

- To solve the SFC migration problem, we firstly model the migration problem of SFCs as the integer linear programming.
- To reduce the reconfiguration cost, the migration time and downtime of SFCs, and improve the remapping success ratio of the SFC, we propose two SFC migration strategies: the minimum number of VNFs migration strategy and the two-step migration strategy.
- In the two-step migration strategy, we make use of the pre-copy based parallel migration strategy for migrating these remapped VNFs in the first step migration, and we make use of the post-copy based parallel migration strategy for migrating these remapped VNFs in the second step migration. And we numerically analyzed the migration time and downtime.
- According to the two-step migration strategy, we have designed a two-step migration algorithm to migrate SFCs.
- We use the federated environment of the fog computing and the cloud computing to emulate and evaluate our proposed algorithm.

ABSTRACT

Network Function Virtualization (NFV) provides a good paradigm for sharing the resources of the physical network. The deployment problem of Service Function Chains (SFCs) composed of a specific order of Virtual Network Functions (VNFs) has become the focus of research. Moreover, to solve the facing challenges of the centralized cloud computing, the researchers have proposed the distributed fog computing. When the mobile user moves among different fog-based radio access networks, the SFC must be migrated. Therefore, in the paper, we research the problem of SFCs migration/remapping caused by the user movement in cloud-fog computing environments. We firstly model the migration problem of SFCs as an integer linear program; then we propose two SFC migration strategies: the minimum number of VNFs migration strategy and the two-step migration strategy, to reduce the reconfiguration cost, the migration time and downtime of SFCs and improve the remapping success ratio of SFCs; and we have designed a two-step migration algorithm to migrate SFCs. We use the cloud-fog computing environment to evaluate our proposed algorithm. The reconfiguration cost, the remapping success ratio, the migration time and the downtime of our proposed algorithms are more excellent than that of benchmark algorithm.

Keywords

Network Function Virtualization; Cloud-Fog computing; Service Function Chain; Live Migration

1. Introduction

In recent years, to solve the rigidity problem of traditional networks, researchers have proposed network virtualization technology [1-3]. With the development of the network virtualization technology, researchers have proposed Network Function Virtualization (NFV) [4-6] technology, through network function virtualization technology, physical resources can be virtualized into Virtual Network Functions (VNFs) [7-9], while virtual network functions can be isolated from each other, to instead of traditional network functions implemented by the specific hardware, so as to reduce the deployment of the special hardware, improve network flexibility and reduce network operating costs.

The specific number and order of virtual network functions form a Service Function Chain (SFC) to support and deal with the user's network traffic, so as to realize the communication and the demand of the user [10, 11]. For example, to satisfy the user's security demand, the SFC may be sequentially composed of User → virtual Firewall (vFW) → virtual Deep Packet Inspection (vDPI) → Terminal. In order to achieve the user's different strategies, SFC has its specific composition. To achieve the user communication, it is very important to deploy these SFCs into the physical network. At present, there have been some researches on the SFC deployment [12-14]. In [12], to improve the network resource utilization, the authors propose an algorithm that takes into account the utilization of the physical link and the physical server. To minimize the total bandwidth consumption, the research [13] proposes a method to jointly design and map multiple SFCs. In order to meet the scalability and the privacy problem of network services, and reduce the deployment complexity of the distributed service chain, the authors exploit the non-cooperative game theory to deploy the distributed SFC and implement the privacy protection [14].

However, as the user increases in geometry, the facing challenges of cloud computing are also increasing, in particular, the network traffic is concentrated in the core network, resulting in a large number of the network congestion and in a long network delay. To solve the facing challenges of the centralized cloud computing, the researchers have proposed the distributed fog computing to expand the centralized cloud computing [15-20]. Because the distributed fog computing environment has a small amount of computing and storage resources, the fog computing environment can perform some delay-sensitive services, and the utilizing of cloud computing resources and fog computing resources at the same time can reduce the network congestion and the energy consumption. Therefore, the fog computing has become a hot research direction for the radio access

* *Corresponding author.*
E-mail address: dliao.uestc@gmail.com (D. Liao).

network, wireless access network, vehicular network and internet of things. Besides, because the distributed fog computing environment has the resources of computing and storage, we can deploy SFC by using the federated environment of the cloud computing and the fog computing, and there are some related researches [21, 22].

With the development of NFV technology, the VNF/SFC migration has gradually become a new research direction. For instance, the authors in [23] first proposed a SFC deployment algorithm to initial deploy SFC, and then in order to save bandwidth resources and reduce the energy consumption, put forward a VNF consolidation and migration algorithm. Similarly, in order to save energy, the research [24] proposes a VNF consolidation and migration algorithm to shutdown servers during low traffic. In [25], to save the energy of the data center, the authors propose a linear programming model to solve this problem. The researches [23-25] are only to adjust these deployed SFCs, they don't take into account the scene that the SFC must be migrated. The research [26] considers migrating the entire Virtual Data Center (VDC) when the servers of the data center need to be maintained or failed, and proposes a VDC migration algorithm, but the proposed algorithm is suitable to the VDC migration, in the SFC migration scene, it does not provide good performance. And although [23-25] research the problem of the VNF consolidation and migration, they did not consider such a scene: in the Fog Radio Access Network (FRAN) environment, due to the mobility of mobile users, when the mobile user moves from a fog radio access network to another fog radio access network, the service provider must migrate these related SFCs to maintain user connectivity, otherwise, the user's communication will be disconnected. Moreover, the SFC migration/remapping algorithm will not only determine the reconfiguration cost and the remapping success ratio, but also affect the migration time and downtime of these SFCs, so the SFC migration algorithm is critical to the migration performance. However, the existing research can't solve the problem well, thus the SFC migration issue is worthy of further studying.

Therefore, in this paper, to solve the SFC migration problem caused by the user movement, we study the problem of the SFC migration in the federated environment of the cloud computing and the fog computing to optimize the migration performance. The main contributions of this work are as follows:

- To solve the SFC migration problem, we firstly model the migration problem of SFCs as the integer linear programming.
- To reduce the reconfiguration cost, the migration time and downtime of SFCs and improve the remapping success ratio of the SFCs, we propose two SFC migration strategies: the minimum number of VNFs migration strategy and the two-step migration strategy.
- In the two-step migration strategy, we make use of the pre-copy based parallel migration strategy for migrating these remapped VNFs in the first step migration, and we make use of the post-copy based parallel migration strategy for migrating these remapped VNFs in the second step migration. And we numerically analyzed the migration time and downtime.
- According to the two-step migration strategy, we have designed a two-step migration algorithm to migrate SFCs.
- We use the federated environment of the fog computing and the cloud computing to emulate and evaluate our proposed algorithms.

The rest of this work is arranged as follows: in section 2, we introduce the related works; we model the migration problem in section 3; we propose the two-step migration algorithm in section 4; section 5 introduces the simulation environment, gives simulation results and analyzes them; finally, in section 6, we conclude this paper.

2. Related Work

2.1 Fog Computing

To solve the facing challenges of the cloud computing, Cisco proposed the fog computing to expand the centralized cloud computing. Due to the advantages of fog computing, the researchers have done some researches [17-22, 27-31], and the fog computing has become a hot research direction for the radio access network, wireless access network, vehicular network and internet of things.

In [17], in order to handle these challenges caused by fog computing, the authors presented the three-layer hierarchical game framework to manage network resources. To solve the security problem of fog computing, the research [18] proposes an architecture framework to guarantees that the user information will not be leaked when the channel is attacked. The research [19] gives a general answer to the ten hot issues of fog computing, such as what is fog computing, what is the relationship between fog computing and cloud computing, what are the scenarios for fog computing, and so on. In order to reduce the network latency, the research [21] uses the mobile edge network to deploy some VNFs of the service function chain.

In [22], the authors studied the fusion of NFV, 5G and fog computing, and proposed a MANO-based architecture to achieve a unified management of internet of things. The research [27] discusses the influence of fog computing on 5G radio access network, and proposes a 5G radio access network based on fog computing. In order to improve quality of experience, the research [28] proposes internet access networks architecture based on fog computing to deploy virtual machines into the user's neighborhood. To deal with the challenges of user growth, in [29], the authors have proposed a radio access networks architecture to provide services, which is based on fog computing and SDN.

To improve the efficiency of face recognition and reduce network transmission, the authors present a face recognition system based on fog computing in Internet of Things [20]. In [30], the authors studied the utilizing of fog computing and SDN to provide services in vehicular networks, in order to overcome the instability of fog communication, a method is proposed to reduce the overhead of control information by using network information. In order to accommodate the increase of vehicle traffic and reduce the delay, the research

[31] proposed a vehicular network architecture to achieve mobile computing.

These researches [17-20, 27-31] on fog computing do not take into account the VNF deployment or migration scenarios, hence they can't be applied to the VNF deployment or migration scenarios. Although [21, 22] combine with fog computing and NFV to conduct research, but they did not study the problem of the VNF/SFC migration.

2.2 SFC Deployment and Migration

With the development of NFV technology, the deployment of SFC has received extensive attention from industry and academia, therefore, the problem deployment of SFC has become a hot research, and has a lot of researches [4-14, 32-39].

To decrease the deployment cost, in [4], the authors studied the deployment of VNF, and presented three algorithms to maximize the profits of service providers. To deal with the challenges of NFV orchestration, the research [5] proposes a NFV management and orchestration architecture to handle dynamic SFC requests. To meet the availability of services and reduce the deployment cost, the authors proposed the QoS-based LP model and the heuristic algorithm to deploy VNF [8]. In [9], the authors study the optimal placement of VNF by considering different VNF traffic changing effects and dependency relations, and formulate the traffic aware placement problem of interdependent VNFs as a graph optimization problem with the objective to load-balance the network, then propose a VNF deployment algorithm to deploy VNFs. To improve the service provider's revenue, in [30], the authors presented the ILP model and the heuristic algorithm to deploy SFC. In [33], the authors present an optimal algorithm and two approximation algorithms to deploy VNFs to minimize deployment costs of SFC requests and maximize the acceptance ratio of SFC requests.

In [10], the authors first build a set of SFCs into a SFC graph, then deploy VNFs in the SFC graph according to the dependencies between VNFs to reduce the bandwidth consumption. In [12], the authors propose an algorithm to deploy SFC to reduce the resource consumption of the physical network, which takes into account the utilization of the physical link and the physical server. In [13], to minimize the total bandwidth consumption, the authors proposed a method to jointly design and map multiple SFCs. To decrease the energy consumption of the data center, the research [34] proposed an energy-aware algorithm to deploy SFC. To reduce the bandwidth and energy consumption, the research [35] puts forward a sampling-based Markov approximation (MA) method, then, to shorten the convergence time of the MA method, the authors find an efficient VNF deployment solution by combining the MA method with matching theory.

In [14], the authors exploit the non-cooperative game theory to deploy the distributed SFC and implement the privacy protection to protect the privacy of the user and reduce the deployment complexity of the distributed service chain. To solve the security requirement of different users, the author proposes network security defense patterns framework to deploy VNFs to meet various security requirements [36]. In order to reduce the transmission and processing delay of VNF, in [37], the authors proposed a genetic algorithm to deploy VNFs, thereby reducing the scheduling time of the entire VNF and increasing the revenue of the service provider. To provide sufficient flexibility for content delivery network, the authors in [38] researched the deployment problem of VNFs in content delivery network, and proposed a linear programming model for deploying SFC. In [39], the authors studied the problem of sharing pipelines in NFV environment, and proposed optimization algorithms for different input assumptions to minimize the delay.

With the number of users continue to increase, to save energy and save bandwidth resources and other goals, the VNF/SFC migration has gradually become a new research direction, and some studies have been devoted to the VNF/SFC migration [23-26, 40-43]. The authors in [23] first proposed a SFC deployment algorithm to initial deploy SFC, and then in order to save bandwidth resources and reduce the energy consumption, presented a VNF consolidation and migration algorithm. To save energy, the research [24] proposes a VNF consolidation and migration algorithm to shutdown servers during low traffic. In [25], the researchers present a linear programming model to solve this problem to save the energy of the data center. To alleviate the problem of packet loss in migration, the research [40] proposed a new interface to achieve the seamless migration of VNF. In [41], to save computing resources and bandwidth resources, the authors proposed a virtual edge architecture with wavelength ADM to realize the live migration of VNF. The research [42] aims to reduce the migration time of VNF, and proposes two heuristic algorithms to migrate the entire virtual machine hosting VNF for different scenarios. The research [43] focuses on the optimal migration problem of VNF when the resource of VNF changes, and proposes a heuristic algorithm to implement the migration of VNF.

Although [23-25, 40-42] research the problem of the VNF consolidation and migration, they don't take into account the scene that the SFC must be migrated due to the mobile user moves from a fog radio access network to another fog radio access network. Although the research [43] focuses on the optimal migration problem of VNF when the resource of VNF changes, it is only to optimize the migration time, it does not take into account the reconfiguration cost, the remapping success ratio and the downtime of the SFC, and it also does not consider migrating SFC in the fog access network. Although the research [26] proposes a VDC migration algorithm to migrate the entire VDC when the servers of the data center need to be maintained or failed, but the proposed algorithm is suitable to the VDC migration, it does not provide good performance for the SFC migration. Hence, the SFC migration issue is worthy of further studying.

3. Problem Description and Formulation

3.1 Problem Description

In this work, we study the problem of SFC migration/ remapping in the cloud-fog network environment when the mobile user moves to different regions. In the fog radio access network environment due to the mobility of mobile users, when the mobile user moves from a fog radio access network to another fog radio access network, the service provider must migrate these related SFCs to maintain user connectivity, otherwise, the user's communication will be disconnected. Moreover, the SFC migration/remapping algorithm will not only determine the reconfiguration cost and the remapping success ratio, but also affect the migration time and downtime of SFCs, so the SFC migration algorithm is critical to the migration performance. Therefore, when the SFC migration requests arrive dynamically, given the locations of the users after the move, the initial mapping solutions of these SFC migration requests and a physical network consisting of the cloud computing environment and the fog radio access network environment, the problem is how to efficiently remap and migrate these SFC migration requests, such that the reconfiguration cost, the blocking ratio, the migration time and the downtime of the SFC are minimized.

3.2 Physical Network

In this paper, the physical network is composed of the cloud network (the cloud computing environment) and multiple fog radio access networks, as shown in Fig.1, and the physical network can be described as an undirected weighted graph $G^P = (N^P, E^P)$. Where $N^P=\{n_1, n_2,\ldots, n_{|NP|}\}$ denotes the set of the physical nodes, and $E^P=\{l_1, l_2,\ldots, l_{|EP|}\}$ depicts the set of the physical links, $|NP|$ and $|EP|$ respectively depict the numbers of the physical nodes and the physical links.

Physical network resource constraint: $SC = (C^N, C^E, L^N)$ is defined as the physical network resource constraints.

Physical node resource attributes: C^N represents the set of the physical node resource attributes, which includes the capacity of the physical node resources $c(n_i)$ and the cost of the per unit physical node resource $p(n_i)$.

Physical link resource attributes: C^E denotes the set of the physical link resource attributes, which includes the capacity of the physical link resources $c(l_i)$ and the cost of the per unit physical link resource $p(l_i)$.

Physical nodes location constraint: L^N depicts the set of the location constraints of the physical network nodes.

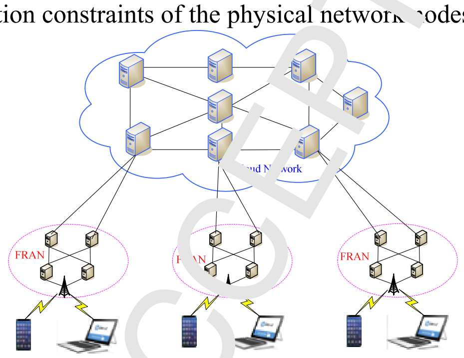


Fig.1 A physical network

3.3 SFC migration request

In this paper, the SFC migration request can be described as an undirected weighted graph $G_F = (N_F, E_F)$, where $N_F= \{f_1, f_2,\ldots, f_{|NF|}\}$ represents the set of VNFs in the SFC migration request, and $|NF|$ depicts the number of VNFs of the SFC request. $E_F=\{e_1,e_2,\ldots,e_{|EF|}\}$ represents the set of SFC links in SFC migration request, and the number of SFC links is $|EF|$.

Migration constraint: we define $MC = (C_N, C_E, V_N, B, IM_N, IM_E, L_N, L_U, L_T)$ as the set of the migration constraints of the SFC migration request.

VNFs resource constraint: $C_N=\{\varepsilon(f_1), \varepsilon(f_2),\ldots, \varepsilon(f_{|NF|})\}$ depicts the set of the node resource demands of these VNFs.

Links resource constraint: $C_E=\{\varepsilon(e_1), \varepsilon(e_2),\ldots, \varepsilon(e_{|EF|})\}$ depicts the set of the link resource demands of these SFC links.

VNFs memory constraint: $V_N=\{V(f_1), V(f_2),\ldots, V(f_{|NF|})\}$ denotes the set of the memory sizes of all VNFs.

VNFs initial mapping records: $IM_N=\{IM\,(f_1), IM\,(f_2),\ldots, IM\,(f_{|NF|})\}$ represents the set of initial mapping solutions of these VNFs.

Links initial mapping records: IM_E represents the set of initial mapping solutions of these SFC links.

VNFs migration bandwidth constraint: B represents the total migration bandwidth requirement of the SFC migration request, and all VNFs of the SFC migration request together share the migration bandwidth resources.

VNFs user and service terminal location constraint: $L_N = \{L(f_1), L(f_2),\ldots, L(f_{|NF|})\}$ denotes the set of the location constraints of all VNFs in the SFC migration request after the user has moved. L_U expresses the location of the user after the user has moved. L_T depicts the location of the service terminal.

Note that, we studied the problem of the SFC migration based on the cloud-fog network environment. The fog radio access network has a small amount of service resources, similar to the cloud computing environment, we can also use NFV technology to share these resources, so we can use the VNFs to replace the hardware network functions in the traditional access network (such as the Packet Data Network Gateway (PGW) and the Serving Gateway (SGW)) to reduce the cost of deploying and maintaining the specialized hardware and increase the flexibility of the network. Hence, in this paper, the VNFs f_1 and f_2 respectively denote the virtual SGW (vSGW) and the virtual PGW (vPGW), and their location constraints should be in the corresponding fog access network. Other VNFs denote the VNFs in the cloud environment (such as vFW and vDPI), and their location constraints should be in the cloud computing environment. A SFC migration request is shown in Fig.2.

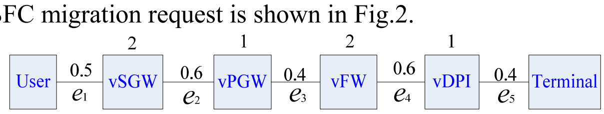


Fig.2 A SFC migration request

3.4 Integer Linear Programming for SFC Remapping

In the reconfiguration process of the SFC request, we focus on how to effectively remap the VNFs and the SFC links, and we first formulate the reconfiguration process of the VNFs as follows:

$$RM_N:(N_F,C_N)\xrightarrow{RM_N}(N^{P1},C^{N1}),$$

$$RM(f_i)\in N^{P1},\forall f_i\in N_F,$$

$$A(RM(f_i)) \geq \varepsilon(f_i), \forall f_i \in N_F,$$

$$Z(f_i, y) \in \{0,1\}, \forall f \in N_F, \forall y \in \{0,1,...,Y\},$$

$$L(RM(f_i)) \in \{0,1,2,...,Y\}, \forall f_i \in N_F,$$

$$Z(f_i, L(RM(f_i))) = 1, \forall f_i \in N_F,$$

Where $N^{P1} \subset N^P$ denotes a subset of the physical nodes re-hosting the VNFs, $C^{N1} \subset C^N$ denotes the physical node resources assigned to the SFC request. $RM(f_i)$ represents a new physical node re-hosting the VNF f_i. $A(RM(f_i))$ indicates available resources of the new physical node $RM(f_i)$. $y \in 0, 1, \ldots, Y$ depicts the number of the network area, $Z(f_i, y) \in \{0,1\}$ is a binary variable, if $Z(f_i, y)=1$ depicts that the *i-th* VNF f_i can be redeployed into the network area, otherwise $Z(f_i, y)=0$. $L(RM(f_i))$ expresses the numbered of the network area of the new physical node $RM(f_i)$. $Z(f_i, L(RM(f_i)))=1$ depicts that the new physical node $RM(f_i)$ meets the location constraint of the *i-th* VNF f_i, otherwise $Z(f_i, L(RM(f_i)))=0$.

In this paper, while remapping the VNFs, we need to remap the SFC links, and the remapping process of the SFC links can be described as follows.

$$RM_E : (E_F, C_E) \xrightarrow{RM_E} (P^1, C^{E1})$$

$$RM(e_i) = p_{e_i}, \quad \forall e_i \in E_F, \exists p_{e_i} \in P^1$$

$$B(p_{e_i}) = \min_{l_j \in p_{e_i}} \{b(l_j)\} \geq \varepsilon(e_i), \quad \forall p_{e_i} \in P^1$$

Where, $P^1 \subset P$ depicts a subset of all physical paths. C^{E1} depicts the bandwidth resources assigned to the SFC request. $RM(e_i)$ and p_{ei} also express a new physical path re-hosting the SFC link e_i. $B(p_{ei})$ indicates available resources of the new physical path p_{ei}.

Therefore, in this paper, while remapping the VNFs, we need to remap the SFC links. Hence, we can build the remapping of the SFC request as the integer linear programming (1).

$$\min\Big(\sum_{f_i \in N_F} P(RM(f_i))\varepsilon(f_i) + \sum_{e_i \in E_F} \sum_{l_j \in p_{e_i}} P(l_j)\varepsilon(e_i)\Big), \qquad (1)$$

$$s.\ t.$$

$$A(RM(f_i)) \geq \varepsilon(f_i), \forall f_i \in N_F$$

$$Z(f_i, y) \in \{0,1\}, \forall f \in N_F, \forall y \in \{0,1,...,Y\}$$

$$L(RM(f_i)) \in \{0,1,2,...,Y\}, \forall f_i \in N_F$$

$$Z(f_i, L(RM(f_i))) = 1, \forall f_i \in N_F$$

$$RM(e_i) = p_{e_i}, \quad \forall e_i \in E_F, \exists p_{e_i} \in P^1$$

$$B(p_{e_i}) = \min_{l_j \in p_{e_i}} \{b(l_j)\} \geq \varepsilon(e_i), \quad \forall p_{e_i} \in P^1$$

3.5 SFC Migration Strategies

In this paper, due to the mobility of mobile users, when the mobile user moves from a fog radio access network to another fog radio access network, we must migrate the SFC request to maintain user connectivity. During the reconfiguration and migration process of the SFC request, we use different migration strategies for different performance requirements.

(1) *The minimum number of VNFs migration strategy*

Although we must migrate the SFC request to maintain user connectivity when the mobile user moves from a fog radio access network to another fog radio access network. To quickly restore the user's service, we may only migrate the least VNFs in the SFC request. Since, in this paper, the VNFs f_1 and f_2 respectively denote the vSGW and the vPGW, and they must be placed in the fog access network where the user is located. So we propose a strategy for minimizing the number of VNFs migration. In the minimum number of VNFs migration strategy, we only remap the vSGW and the vPGW to quickly restore the user's service and migrate these remapped VNFs by using the pre-copy based parallel migration strategy, it is described in Ref. [26, 44, 45]. The advantage of the strategy is to minimize the running time of the remapping algorithm, the migration time and the downtime of the SFC request. But when the user migrates to a distant location through multiple moves, the strategy will cause that the length of remapping path and the reconfiguration cost are very high, resulting in a waste of physical resources and increasing the blocking ratio of the SFC request. In such a situation, the minimum number of VNFs migration strategy is not conducive to saving physical resources, reducing the reconfiguration cost and the blocking ratio of the SFC request. And we can use the two-dimensional discrete random walk process to prove that no matter where the starting position of the user is, but with the user moves, the user can reach any one fog access network, that will cause the user to migrate to a distant location.

We model the coverage of the fog access networks as a two-dimensional plane, and each fog access network is a discrete point in the two-dimensional plane, and the user random walks in the two-dimensional discrete plane. So we can model the movement process of the user as the two-dimensional discrete random walk process. We assume that the initial position of the user is at the coordinate origin, the user walks S steps and arrives the point (x, y), i.e., the total number of steps is S, where the user walks A steps along the X axis and walks B steps along the Y axis. In the X axis, the user can respectively walk to the right and the left, and the probability of walking to the right is p_1 and the probability of walking to the left is p_2. In the Y axis, the user can respectively walk to the above and the down, and the probability of walking to the above is p_3 and the probability of walking to the down is p_4, and $p_1+p_2+p_3+p_4=1$, $p_1>0$, $p_2>0$, $p_3>0$, $p_4>0$. And each walk is independent of each other. So, we can use two superposed Bernoulli models to solve the two-dimensional discrete random walk process.

Event C: the user walks S steps and arrives the point (x, y), we assume that $x>0$, $y>0$, $|x|+|y| \leq S$, so

$$\because A + B = S, \therefore x + y \leq S, \therefore x \leq S - y.$$

We use x_i to denote the coordinate of the user after walk i steps along the X axis. So we have:

$$x_i = x_{i-1} + (-1)^{\delta_i}.$$

Where δ_i is equal to 0 when the i-th step moves to the right, δ_i is equal to 1 when the i-th step moves to the left. So if the user walks A steps along the X axis and arrives x, we have

$$x_A = \sum_{i=1}^{A} (-1)^{\delta_i} = \alpha - \beta .$$

Where α represents the number of $\delta_i = 0$, β denotes the number of $\delta_i = 1$. So we have: $\alpha - \beta = x$, $\alpha + \beta = A$, $A - x = 2\beta$, so $A \equiv x \pmod 2$. So if the user walks A steps along the X axis and arrives x, we have $A \equiv x \pmod 2$.

In the same way, we can prove that if the user walks B steps along the Y axis and arrives y, we have $B \equiv y \pmod 2$. So if the user walks A steps along the X axis and the user walks B steps along the Y axis and arrives the point (x, y), we can obtain $S \equiv x+y \pmod 2$. So when $S \not\equiv x+y \pmod 2$, in this case, the point (x, y) is unreachable, so $P(C)=0$. When $S \equiv x+y \pmod 2$, in this case, the point (x, y) is reachable, and there are three cases:

(1) When $A=x$, $B=S-x$.

E_1: The user walks x steps along the X axis and walks $S-x$ steps along the X axis, so

$$P(E_1) = C_S^x (p_1 + p_2)^x (p_3 + p_4)^{S-x} .$$

F_1: The user walks x steps along the right of the X axis and arrives x, so

$$P(F_1 \mid E_1) = C_x^x p_1^x p_2^0 = p_1^x .$$

G_1: The user walks $S-x$ steps along the Y axis and arrives y, we assume that the user walks up i steps, so

$$i - (S - x - i) = y \Rightarrow i = \frac{S - x + y}{2} ,$$

$$P(G_1 \mid E_1) = C_{S-x}^i p_3^i p_4^{S-x-i} = C_{S-x}^{\frac{S-x+y}{2}} p_3^{\frac{S-x+y}{2}} p_4^{\frac{S-x-y}{2}} ,$$

$$\begin{aligned} P(C) &= P(E_1 F_1 G_1) \\ &= P(E_1) P(F_1 G_1 \mid E_1) = P(E_1) P(F_1 \mid E_1) P(G_1 \mid E_1) \\ &= [C_S^x (p_1 + p_2)^x (p_3 + p_4)^{S-x}][C_{S-x}^{\frac{S-x+y}{2}} p_3^{\frac{S-x+y}{2}} p_4^{\frac{S-x-y}{2}}] p_1^x \end{aligned} .$$

(2) When $A=S-y$, $B=y$.

E_2: The user only walks y steps along the Y axis and walks $S-y$ steps along the X axis, so

$$P(E_2) = C_S^y (p_1 + p_2)^{S-y} (p_3 + p_4)^y .$$

F_2: The user walks $S-y$ steps along the X axis and arrives x, we assume that the user walks j steps along the right of the X axis, so

$$j - (S - y - j) = x \Rightarrow j = \frac{S + x - y}{2} ,$$

$$P(F_2 \mid E_2) = C_{S-y}^j p_1^j p_2^{S-y-j} = C_{S-y}^{\frac{S+x-y}{2}} p_1^{\frac{S+x-y}{2}} p_2^{\frac{S-x-y}{2}} ,$$

G_2: The y steps are along the above of the Y axis and arrives y, so

$$P(G_2 \mid E_2) = C_y^y p_3^y p_4^0 = p_3^y ,$$

$$\begin{aligned} P(C) &= P(E_2 F_2 G_2) \\ &= P(E_2) P(F_2 G_2 \mid E_2) = P(E_2) P(F_2 \mid E_2) P(G_2 \mid E_2) \\ &= [C_S^y (p_1 + p_2)^{S-y} (p_3 + p_4)^y][C_{S-y}^{\frac{S+x-y}{2}} p_1^{\frac{S+x-y}{2}} p_2^{\frac{S-x-y}{2}}] p_3^y \end{aligned} .$$

(3) When $x \le A \le S-y$, $y \le B \le S-x$.

E_3: The user only walks A steps along the X axis and walks B steps along the Y axis, so

$$P(E_3) = C_S^A (p_1 + p_2)^A (p_3 + p_4)^{S-A} .$$

F_3: The user walks A steps along the X axis and arrives x, we assume that the user walks a steps along the right of the X axis, so

$$a - (A - a) = x \Rightarrow a = \frac{A + x}{2} ,$$

$$P(F_3 \mid E_3) = C_A^a p_1^a p_2^{A-a} = C_A^{\frac{A+x}{2}} p_1^{\frac{A+x}{2}} p_2^{\frac{A-x}{2}} ,$$

G_3: The user walks B steps along the Y axis and arrives y, we assume that the user walks b steps along the above of the Y axis, so

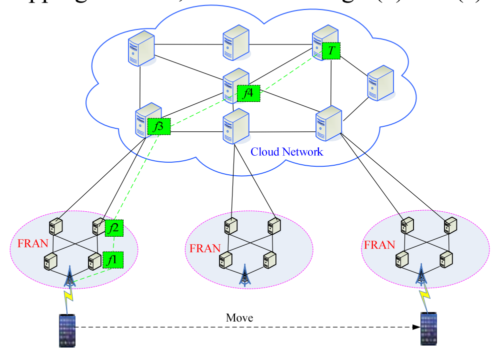
$$b - (B - b) = y \Rightarrow b = \frac{B + Y}{2} ,$$

$$\begin{aligned} P(G_3 \mid E_3) &= C_B^b p_3^b p_4^{B-b} = C_{S-A}^{\frac{S-A+y}{2}} p_3^{\frac{S-A+y}{2}} p_4^{\frac{S-A-y}{2}} , \\ P(C) &= P(E_3 F_3 G_3) = P(E_3) P(F_3 G_3 \mid E_3) \\ &= P(E_3) P(F_3 \mid E_3) P(G_3 \mid E_3) \\ &= [C_S^A (p_1 + p_2)^A (p_3 + p_4)^{S-A}][C_A^{\frac{A+x}{2}} p_1^{\frac{A+x}{2}} p_2^{\frac{A-x}{2}}] \\ &\quad \times [C_{S-A}^{\frac{S-A+y}{2}} p_3^{\frac{S-A+y}{2}} p_4^{\frac{S-A-y}{2}}] \end{aligned} .$$

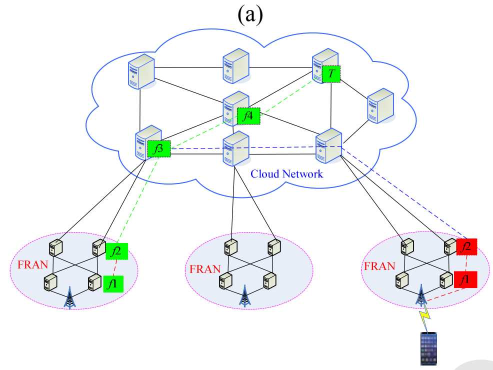
As can be seen from the above proof, the probability of Event C $P(C)$ is greater than 0 as long as the constraints $x>0$, $y>0$, $|x|+|y| \le S$ and $S \equiv x+y \pmod 2$ are satisfied, i.e., the user can reach all points $x>0$, $y>0$ on the plane. In the same way, we can prove that the user can reach all points on the plane when the constraints $x \in Z$, $y \in Z$, $|x|+|y| \le S$ and $S \equiv |x|+|y| \pmod 2$ are satisfied. So our hypothesis is reasonable that the user can migrate to a distant location through multiple moves.

(2) *The two-step migration strategy for the SFC*

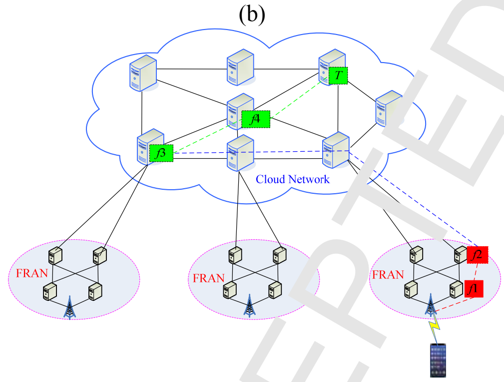
To improve the minimum number of VNFs migration strategy, based on the minimum number of VNFs migration strategy, we present a two-step migration strategy for the SFC. The execution of the two-step migration strategy is illustrated by Fig.3, where Fig.3(a) describes an initial mapping solution of the SFC request, Fig.3(b), (c), (d) and (e) describe the remapping/migration process. In the two-step migration strategy for the SFC, the first step migration is to remap the minimum number of the VNFs, and migrate these remapped VNFs by using the pre-copy based parallel migration strategy, it is described in Ref. [24, 42, 43]. When these remapped VNFs are completely migrated, we connect the solution of the remapping part with the solution of the non-remapping part to form a temporary migration solution to quickly restore the user's service, as shown in Fig.3(b), then cancel the initial mapping solution of the migration part, as shown in Fig.3(c). Moreover, in the first step migration, if we only migrate the vSGW and the vPGW, the solution of the remapping part may not be connected to the solution of the non-remapping part, so to restore the user's services, we may need to migrate more than two VNFs. The second step migration is to remap and migrate other VNFs to save physical resources and reduce the reconfiguration cost, as shown in Fig.3(d) and (e), and migrate other VNFs by using the post-copy based parallel migration strategy. In the second step migration, we have been using the temporary migration solution to provide service to the user, and until the migration of other VNFs have been completed, we have completely cancelled the initial mapping solution, as shown in Fig.3(d) and (e).



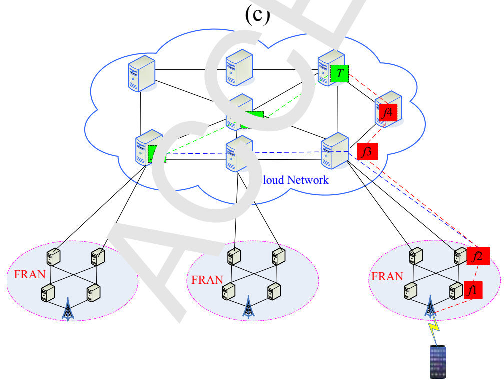
(a)



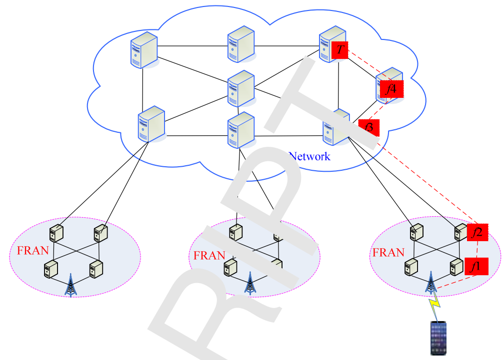
(b)



(c)



(d)



(e)

Fig. 3 The two step migration strategy

3.6 SFC Migration Time and Downtime

In the first step migration of the two-step migration strategy for the SFC, we migrate these remapped VNFs by using the pre-copy based parallel migration strategy, and we assume that the migration number of VNFs is F_f, and F_f must be greater than or equal to 2, i.e., $F_f >=2$. In the first step migration, in terms of a single VNF, we make use of the pre-copy migration mechanism for migrating a VNF. The migration time and the downtime of the single VNF are described in Ref. [26, 44, 45]. The migration time of the *i-th* VNF f_i can be computed as follow.

$$T_{i,mig} = \sum_{j=1}^{\lambda_i+1} T_{i,j} = \frac{V(f_i)}{B(f_i)}\frac{1-r_i^{\lambda_i+1}}{1-r_i}, i=1,2,...,F_f \quad (2)$$

$$\lambda_i = \min\{\lceil \log_{r_i}(V_{th}/V(f_i))\rceil, \lambda_{\max}\} \quad (3)$$

Where λ_i depicts the actual number of iterations that must be less than the maximum number of iterations λ_{max}, $T_{i,j}$ depicts the *j-th* iterative migration time of the memory of the *i-th* VNF f_i. We define V_{th} as the stop-iteration threshold. $B(f_i)$ denotes the obtained migration rate of the *i-th* VNF f_i. $r_i = PD/B(f_i)$ denotes the ratio of the dirtying rate to the migration rate, where D and P respectively depict the dirtying rate of memory page and memory page size.

When the λ_i-*th* iteration is complete, the *i-th* VNF f_i will stop working, so the starting point of the downtime $T_{i,down}^{start}$ can be computed as:

$$T_{i,down}^{start} = \sum_{j=1}^{\lambda_i} T_{i,j} = \frac{V(f_i)}{B(f_i)}\frac{1-r_i^{\lambda_i}}{1-r_i}, i=1,2,...,F_f. \quad (4)$$

The end point of the downtime $T_{i,down}^{end}$ is the end point of the migration.

$$T_{i,down}^{end} = \sum_{j=1}^{\lambda_i+1} T_{i,j} = \frac{V(f_i)}{B(f_i)}\frac{1-r_i^{\lambda_i+1}}{1-r_i}, i=1,2,...,F_f \quad (5)$$

Hence, the downtime of the *i-th* VNF f_i can be computed as in Equation (6).

$$T_{i,down} = T_{i,down}^{end} - T_{i,down}^{start} = \frac{V(f_i)}{B(f_i)} r_i^{\lambda_i}, i=1,2,...,F_f \quad (6)$$

In terms of multiple VNFs, we adopt the pre-copy based parallel migration mechanism for migrating multiple VNFs, and during the migration process, due to the migrated F_f

VNFs together share the total migration bandwidth, the obtained migration rate of each VNF is B/F_f.

Hence, in the first step migration, the migration time of the first step migration $T_{F,mig}$ is the migration time of the VNF that lastly completes migration in the first step migration.

$$T_{F,mig} = max\{T_{i,mig}\} = max\left\{\frac{V(f_i)}{B/F_f}\frac{1-r_i^{\lambda_i+1}}{1-r_i}\right\}, i=1,2,...,F_f \quad (7)$$

The starting point of the downtime of the first step migration of the SFC migration request $T_{F,down}^{start}$ is the time of the VNF that firstly shutdowns in the first step migration.

$$T_{F,down}^{start} = min\{T_{i,down}^{start}\} = min\left\{\frac{V(f_i)}{B/F_f}\frac{1-r_i^{\lambda_i}}{1-r_i}\right\}, i=1,2,...,F_f \quad (8)$$

The end point of the downtime of the first step migration of the SFC migration request $T_{F,down}^{end}$ is the time of the VNF that lastly completes migration in the first step migration.

$$T_{F,down}^{end} = max\{T_{i,down}^{end}\} = max\left\{\frac{V(f_i)}{B/F_f}\frac{1-r_i^{\lambda_i+1}}{1-r_i}\right\}, i=1,2,...,F_f \quad (9)$$

Hence, the downtime of the first step migration of the SFC migration request $T_{F,down}$ can be computed as follow.

$$\begin{aligned} T_{F,down} &= T_{F,down}^{end} - T_{F,down}^{start} \\ &= max\left\{\frac{V(f_i)}{B/F_f}\frac{1-r_i^{\lambda_i+1}}{1-r_i}\right\} - min\left\{\frac{V(f_i)}{B/F_f}\frac{1-r_i^{\lambda_i}}{1-r_i}\right\}, i=1,2,...,F_f \end{aligned} \quad (10)$$

In the second step migration, we migrate other VNFs by using the post-copy based parallel migration strategy, and we assume that the migration number of VNFs is S_f, due to the migrated S_f VNFs together share the total migration bandwidth, the obtained migration rate of each VNF is B/S_f. In the second step migration, in terms of a single VNF, we adopt the post-copy migration mechanism for migrating a VNF. The migration time of the *j-th* VNF f_j can be computed as follow.

$$T_{j,mig} = \frac{V(f_j)}{B/S_f}, j = F_f+1,...,F_f+S_f \quad (11)$$

In the second step migration, in terms of multiple VNFs, we adopt the post-copy based parallel migration mechanism for migrating multiple VNFs. The migration time of the second step migration $T_{S,mig}$ is the migration time of the VNF that lastly completes migration in the second step migration.

$$T_{S,mig} = max\{T_{j,mig}\} = max\left\{\frac{V(f_j)}{B/S_f}\right\}, j = F_f+1,...,F_f+S_f \quad (12)$$

Due to in the second step migration, the user can get service through the temporary migration solution. So in the second step migration, the SFC migration request will don't stop, and the downtime of the second step migration is zero. So the migration time of the SFC migration request $T_{SFC,mig}$ can be computed as in Equation (13).

$$\begin{aligned} T_{SFC,mig} &= T_{F,mig} + T_{S,mig} \\ &= max\left\{\frac{V(f_i)}{B/F_f}\frac{1-r_i^{\lambda_i+1}}{1-r_i}\right\} + max\left\{\frac{V(f_j)}{B/S_f}\right\}, \\ & i=1,...,F_f, j=F_f+1,...,F_f+S_f \end{aligned} \quad (13)$$

So the downtime of the SFC request $T_{SFC,down}$ can be computed as follow.

$$\begin{aligned} T_{SFC,down} &= T_{F,down} \\ &= max\left\{\frac{V(f_i)}{B/F_f}\frac{1-r_i^{\lambda_i+1}}{1-r_i}\right\} - min\left\{\frac{V(f_i)}{B/F_f}\frac{1-r_i^{\lambda_i}}{1-r_i}\right\}, i=1,...,F_f \end{aligned} \quad (14)$$

Due to the minimum number of VNFs migration strategy only remaps the vSGW and the vPGW and migrates these remapped VNFs by adopting the pre-copy based parallel migration mechanism, in the minimum number of VNFs migration mechanism, the downtime and migration time of the SFC migration request can be computed as in Equation (7) and (14), respectively.

To clearly illustrate the disadvantages and the advantages of the minimum number of VNFs migration strategy and the two-step migration mechanism in the downtime and migration time. We show the numerical results of the minimum number of VNFs migration strategy, the two-step migration mechanism and the pre-copy based parallel migration mechanism in [26], and we respectively describe these three strategies as strategy-1, strategy-2 and strategy-3 in Fig.4 and Fig.5. The key parameters are set as: the memory size of each VNF all is 1GB, the dirtying rate of memory page is 2500pps, the memory page size is 4KB, the maximum number of iterations is 8, moreover, the physical resource capacity is unlimited.

Fig.4 displays the migration times of the minimum number of VNFs migration strategy, the two-step migration strategy and the pre-copy based parallel migration strategy when the number of VNFs in the SFC migration request (i.e., *n*) varies among 5, 6, 7 and 8. From the numerical results of Fig.4, we can see that due to the minimum number of VNFs migration strategy only remaps the vSGW and the vPGW, and the two-step migration strategy and the pre-copy based parallel migration strategy need to migrate all VNFs, the migration time of the minimum number of VNFs migration strategy is shortest in the three strategies; and the migration time of the two-step migration mechanism is lower than that of the pre-copy based parallel migration mechanism due to the second step migration of the two-step migration strategy uses the post-copy based parallel migration strategy that can reduce the total migration time, however the pre-copy based parallel migration strategy migrate all VNFs at the same time.

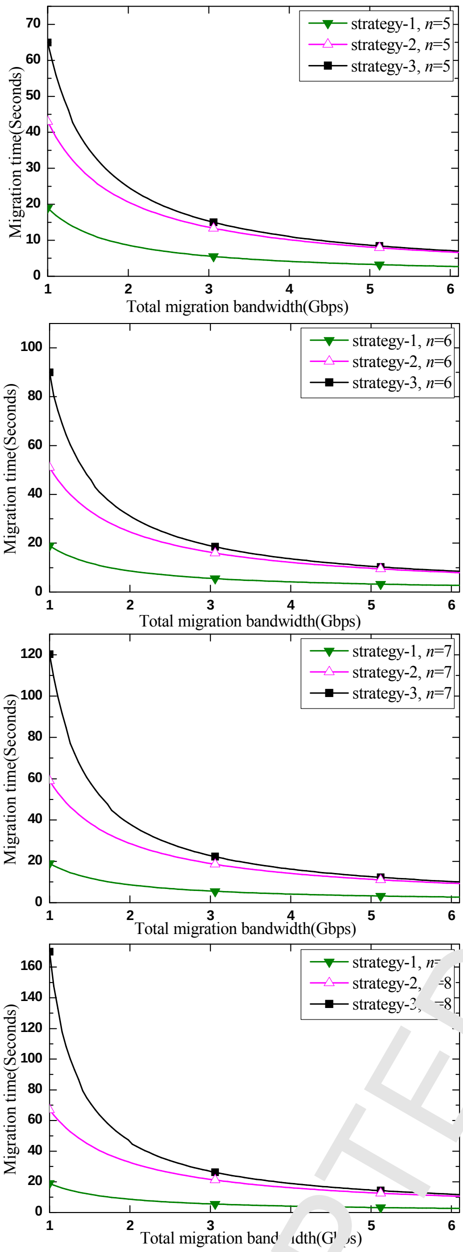


Fig.4 The migration time

Fig.5 reveals the downtime of the minimum number of VNFs migration strategy, the two-step migration strategy and the pre-copy based parallel migration strategy when the number of VNFs in the SFC migration request changes among 5, 6, 7 and 8. From the numerical results, we can see that the curves of the downtimes of the minimum number of VNFs migration strategy and the two-step migration strategy are coincident due to the minimum number of VNFs migration strategy and the first step migration in the two-step migration strategy only remap the vSGW and the vPGW when the physical resource capacity is unlimited, and in the second step migration of the two-step migration strategy, the user can get service through the temporary migration solution. So in the second step migration, the SFC migration request will don't stop, and the downtime of the second step migration is zero. And the downtimes of the minimum number of VNFs migration strategy and the two-step migration strategy are lower than that of the pre-copy based parallel migration mechanism. Moreover, since the characteristics of iterative migration, the downtime is oscillating down as the total migration bandwidth increases.

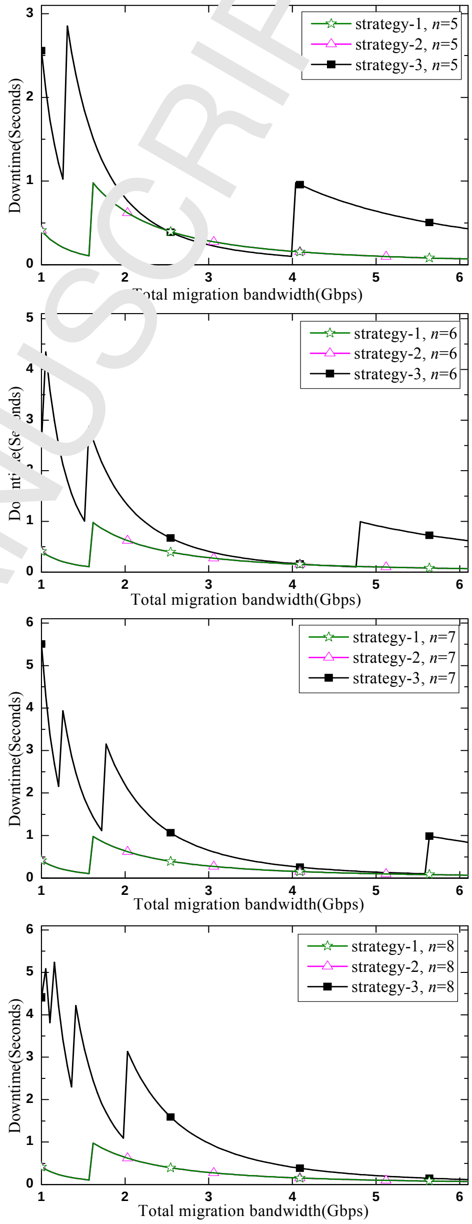


Fig.5 The downtime

4. Heuristic Algorithm

Due to the reconfiguration problem of the SFC migration request in the cloud-fog environment is a NP-hard problem, to achieve the effective reconfiguration solution, based on the two-step migration strategy for the SFC, we propose the SFC two-step migration algorithm (SFCTSM). In the SFC two-step migration algorithm, we suppose that the SFC migration request is dynamically arrived in accordance with the Poisson process, and these SFC migration requests are stored in the queue *ArrivedSFC*. We define a set of blocking requests, described as SFC_{blo}. We define *Tcost* as the total redeployment cost of all SFC requests that are successfully remapped. In the SFC two-step migration algorithm, we migrate the first SFC migration request SFC_1 in the queue *ArrivedSFC* each time, when we migrate the first SFC migration request SFC_1, we first call the FSRMSFC procedure to carry out the first step migration, to remap the minimum number of the VNFs and migrate these remapped VNFs by using the pre-copy based parallel migration strategy and quickly restore the user's service, then call the SSRMSFC procedure to carry out the second step migration, to remap and migrate other VNFs to save physical resources and reduce the reconfiguration cost, and migrate other VNFs by adopting the post-copy based parallel migration mechanism.

Algorithm 1 shows the pseudo-code of the proposed SFC migration algorithm.

***Algorithm* 1:** SFC Two-step Migration (SFCTSM)

Input:1. Physical network $G^P = (N^P, E^P)$ and resource constraints $SC = (C^N, C^E, L^N)$;
2. SFC migration requests queue *ArrivedSFC*.

Output: Total redeployment cost *Tcost* and the set of blocked SFCs, SFC_{blo}.

1: Initialization: **let** F_f=0, *Tcost*=0 and SFC_{blo}=∅;
2: **while** *ArrivedSFC*≠∅, **do**
3: Release the occupied resources of the expired SFC requests;
4: Call FSRMSFC procedure for redeploying the first SFC request SFC_1 in *ArrivedSFC*;
5: **if** found a redeployment solution *RM* for SFC_1 and $1< F_f <|NF|$, **then**
6: Call SSRMSFC procedure for redeploying SFC_1;
7: **end if**
8: **if** SFC_1 is redeployed successfully, **then**
9: updating *Tcost* and the physical network;
10: **else**
11: $SFC_{blo} = SFC_{blo} \cup \{SFC_1\}$;
12: **end if**
13: $ArrivedSFC = ArrivedSFC \backslash \{SFC_1\}$;
14: **end while**
15: **return** *Tcost* and SFC_{blo}.

The FSRMSFC procedure is used to carry out the first step migration. In the FSRMSFC procedure, the vSGW f_1 and the vPGW f_2 must be migrated, and when each VNF f_i, i=2,…, $|N_F|$ is remapped successfully, the FSRMSFC procedure tries to find a temporary path to connect the solution of the remapping part with the solution of the non-remapping part to form a temporary migration solution, if the temporary path is not found, the FSRMSFC procedure remaps the next VNF and tries to find a temporary path to connect the solution of the remapping part with the solution of the non-remapping part, the FSRMSFC procedure always carries out this process until the temporary path is found, then we call the RBAFSM procedure for migrating these remapped VNFs by using the pre-copy based parallel migration strategy to quickly restore the user's service.

The remapping cost of each VNF f_i $CostVNF(f_i \rightarrow n_k)$ can be defined as follows:

$$CostVNF(f_i \rightarrow n_k) = p(n_k)\varepsilon(f_i)\,. \tag{15}$$

The remapping cost of each SFC link e_i $Cost(p_{ei})$ can be computed as in Equation (16).

$$Cost(p_{ei}) = \sum_{l_j \in p_{ei}} p(l_j)\varepsilon(e_i) + Cost(p(n_k, L_T)) \tag{16}$$

Where $p(n_k, L_T)$ is the physical path connecting the physical node n_k and the service terminal, and the bandwidth resource requirement of the physical path $p(n_k, L_T)$ is $\varepsilon(e_{i+1})$, and it can be computed as follows.

$$Cost(p(n_k, L_T)) = \sum_{l_j \in p(n_k, L_T)} p(l_j)\varepsilon(e_{i+1}) \tag{17}$$

The total remapping cost of each VNF f_i $TCostVNF(f_i \rightarrow n_k)$ can be computed as in Equation (18).

$$TCostVNF(f_i \rightarrow n_k) = CostVNF(f_i \rightarrow n_k) + Cost(p_{e_i}) \tag{18}$$

Procedure 1: The First Step for ReMapping a SFC (FSRMSFC)

Input: 1. Physical network $G^P = (N^P, E^P)$ and resource constraints $SC = (C^N, C^E, L^N)$;
2. The first SFC migration request $G_F = (N_F, E_F)$ and migration constraints $MC = (C_N, C_E, V_N, B, IM_N, IM_E, L_N, L_U, L_T)$.

Output: Remapping solution *RM*.

1: **for** each VNF f_i, i=1,2,…, $|NF|$, $f_i \in N_F$, **do**
2: **for** each node $n_k \in N^P$, **do**
3: **if** the node n_k satisfies the location constraint of the VNF f_i, **then**
4: Redeploy f_i into the node n_k, compute $CostVNF(f_i \rightarrow n_k)$ according to Equation (15);
5: Find the shortest paths p_{ei} and $p(n_k, L_T)$, compute $Cost(p_{ei})$ according to Equation (16); compute the total redeployment cost $TCostVNF(f_i \rightarrow n_k)$ according to Equation (18);
6: **end if**
7: **end for**
8: Find the redeployment solution of the VNF f_i with the minimal total redeployment cost $TCostVNF(f_i \rightarrow n_k)$, and store the redeployment solution of f_i in *RM*;
9: **if** $2 \leqslant i < |NF|$, **then**
10: Find the shortest path $p(RM(f_i), IM(f_{i+1}))$;
11: **if** found a shortest path $p(RM(f_i), IM(f_{i+1}))$, **then**
12: Call the RBAFSM procedure for migrating the VNFs f_i, $F_f = i$;
13: **break**;
14: **end if**
15: **end if**
16: **end for**
17: **return** F_f and *RM*.

The SSRMSFC procedure is used to carry out the second step migration. In the SSRMSFC procedure, other VNFs are remapped to save physical resources and reduce the reconfiguration cost, then we call the RBAFSSM procedure for migrating these remapped VNFs by using the post-copy based parallel migration strategy. In the second step migration, we have been using the temporary migration solution to provide service to the user, and until the migration of other VNFs have been completed, we have completely cancelled the initial mapping solution.

***Procedure* 2:** The Second Step for ReMapping a SFC (SSRMSFC)

Input: 1. Physical network $G^P = (N^P, E^P)$ and resource constraints $SC = (C^N, C^E, L^N)$;
2. The first SFC migration request $G_F = (N_F, E_F)$ and migration constraints $MC = (C_N, C_E, V_N, B, IM_N, IM_E, L_N, L_U, L_T)$;
3. The migration number of VNFs in the first step migration F_f and the remapping solution *RM*.

Output: Remapping solution *RM*.

for each VNF f_j, $j= F_f+1,\ldots, |NF|, f_i \in N_F$, **do**
    for each node $n_k \in N^P$, **do**
        if the node n_k satisfies the location constraint of the VNF f_j, **then**
            Redeploy f_j into the node n_k, compute $CostVNF(f_j \rightarrow n_k)$ according to Equation (15);
            Find the shortest paths p_{ei} and $p(n_k, L_T)$, compute $Cost(p_{ei})$ according to Equation (16); compute the total redeployment cost $TCostVNF(f_j \rightarrow n_k)$ according to Equation (18);
        end if
    end for
    Find the redeployment solution of the VNF f_j with the minimal total redeployment cost $TCostVNF(f_j \rightarrow n_k)$, and store the redeployment solution of f_j in *RM*;
end for
call the RBASSM procedure for migrating these VNFs;
return *RM*.

The RBAFSM procedure is responsible for carrying out routing and bandwidth allocation of the first step migration, and computes the downtime and migration time of the first step migration. The RBASSM procedure is responsible for carrying out routing and bandwidth allocation of the second step migration, and computes the downtime and migration time of the second step migration and computes the downtime and migration time of the SFC migration request.

***Procedure* 3:** Routing and bandwidth allocation of the first step migration (RBAFSM)

Input: 1. Physical network $G^P = (N^P, E^P)$ and resource constraints $SC = (C^N, C^E, L^N)$;
2. The first SFC migration request $G_F = (N_F, E_F)$ and migration constraints $MC = (C_N, C_E, V_N, B, IM_N, IM_E, L_N, L_U, L_T)$;
3. The migration number of VNFs in the first step migration F_f and the remapping solution *RM*.

Output: the downtime and migration time of the first step migration of the SFC migration request $T_{F,mig}$, $T_{F,down}$.

Initialization: **let** $T_{i,mig}$=0, $T^{start}_{i,down}$ =0, $T^{end}_{i,down}$ =0, $T_{i,down}$=0, $T_{F,mig}$=0, $T^{start}_{F,down}$ =0, $T^{end}_{F,down}$ =0, $T_{F,down}$=0, i=1,…, F_f;
Find the path $p(RM(f_i), IM(f_i))$ with the minimal cost for each VNF f_i, i=1,…, F_f;
When migration paths of all VNFs , i=1,…, F_f are found, we migrate all VNFs by using the pre-copy based parallel migration strategy;
Compute $T_{i,mig}$, $T^{start}_{i,down}$ and $T^{end}_{i,down}$ of each VNF f_i, according to Equation (2) , (4) and (5), respectively;
Compute $T_{F,mig}$, $T^{start}_{F,down}$, $T^{end}_{F,down}$ and $T_{F,down}$ in Equation (7) , (8) , (9) and (10) , respectively;
return $T_{F,mig}$ and $T_{F,down}$.

***Procedure* 4:** Routing and bandwidth allocation of the second step migration (RBASSM)

Input: 1. Physical network $G^P = (N^P, E^P)$ and resource constraints $SC = (C^N, C^E, L^N)$;
2. The first SFC migration request $G_F = (N_F, E_F)$ and migration constraints $MC = (C_N, C_E, V_N, B, IM_N, IM_E, L_N, L_U, L_T)$;
3. The migration number of VNFs in the first step migration F_f and the remapping solution *RM*.

Output: the downtime and migration time of the SFC migration request $T_{SFC,mig}$, $T_{SFC,down}$.

Initialization: **let** $T_{j,mig}$=0, $T_{S,mig}$ =0, $T_{SFC,mig}$=0, $T_{SFC,down}$=0, $j= F_f+1,\ldots, |NF|$;
Find the path $p(RM(f_j), IM(f_j))$ with minimal cost for each VNF f_j, $j= F_f+1,\ldots, |NF|$;
When migration paths of all VNFs , $j= F_f+1,\ldots, |NF|$ are found, we migrate all VNFs by using the post-copy based parallel migration strategy;
Compute $T_{j,mig}$ of each VNF f_j, according to Equation (11);
Compute $T_{S,mig}$, $T_{SFC,mig}$, and $T_{SFC,down}$ in Equation (12) , (13) , and (14) , respectively;
return $T_{SFC,mig}$ and $T_{SFC,down}$.

5. Simulation Results

5.1 Simulation Environment

In this paper, the physical network is composed of the cloud network (the cloud computing environment) and multiple fog radio access networks. In our simulations, we evaluate the reconfiguration cost of the SFC migration requests, the running time of the algorithm, the downtime and migration time of the SFC migration requests in the scenario of the unlimited resource capacity, and we evaluate the remapping success ratio of the SFC migration requests in the scenario of the limited resource capacity, respectively. In the scenario of the limited resource capacity, we suppose that the resource capacity of the physical nodes obeys the uniform distribution U(20,40), and the resource capacity of the physical links obeys the uniform distribution U(30,50), and the costs of per unit resource of the physical nodes and links are also 1 unit. In our simulations, we measure the performance of our proposed algorithms and the contrast algorithm when the number of VNFs in the SFC migration request varies among 5, 6, 7 and 8, we suppose that the SFC migration requests dynamically arrive in accordance with the Poisson process. In all cases, the resource requirements of VNFs all obey the uniform distribution U(5,10) unit, and the resource requirements of the SFC links all obey the uniform distribution U(5,10) Gbs, the memory size of each VNF all is 1GB, the dirtying rate of memory page is 2500pps, the memory page size is 4KB, the total migration bandwidth requirements of each SFC migration request all are 1Gbs when the number of VNFs in the SFC migration request (i.e., *n*) varies among 5, 6, 7 and 8. The migration location of the user of each SFC migration request is randomly distributed in the fog access networks and the location of the service terminal of each SFC migration request is randomly distributed in the core network.

At present, the researches on VNF mostly consider the VNF deployment, although there are some researches on the problem of the VNF consolidation and migration, they don't take into account the scene that the SFC must be migrated due to the mobile user moves from a fog radio access network to another fog radio access network. So they can't compare with our algorithms. The research [26] considers migrating the entire VDC and proposes the VDC-M algorithm when the servers of the data center need to be maintained or failed, to show the performance of our proposed algorithm, we compare our proposed algorithms with the VDC-M algorithm, moreover, in partial results, we also added the results of the heuristic algorithm based on the minimum number of VNFs migration strategy, SFCMM.

5.2 Simulation Result and Analysis

Fig.6 denotes the total reconfiguration costs of the SFCMM algorithm, the VDC-M algorithm and SFCTSM algorithm when the number of VNFs in the SFC migration request (i.e., *n*) varies among 5, 6, 7 and 8. From the results, we can see that due to the SFCMM algorithm only remaps the vSGW and the vPGW to quickly restore the user's service, when the user migrates to a distant location, the strategy will cause that the total reconfiguration cost is the highest in the three algorithms; moreover, due to the VDC-M algorithm is suitable to the VDC migration, in the SFC migration scene, it does not provide the best performance; due to the SFCTSM algorithm is specifically proposed for the total reconfiguration problem of the SFC migration request in the cloud-fog environment, in term of the total reconfiguration cost, the total reconfiguration cost of the SFCTSM algorithm is the lowest in the three algorithms.

Fig.7 compares the remapping blocking ratio of the VDC-M algorithm and SFCTSM algorithm when the number of VNFs in the SFC migration request varies among 5, 6, 7 and 8. Due to the SFCMM algorithm only remaps the vSGW and the vPGW, then connects the solution of the remapping part with the solution of the non-remapping part to form the final migration solution to quickly restore the user's service. So we must first deploy an initial solution and allocate resources to the initial solution in the simulation, and then migrate these SFC requests successfully deployed the initial solution, which will result in the arrival rate of the SFC migration request successfully deployed the initial solution is low and the blocking ratios of the VDC-M algorithm and SFCTSM algorithm are close to zero and the performance gap between the VDC-M algorithm and SFCTSM algorithm can't be tested. So, Fig.7 only compares the remapping blocking ratio of the VDC-M algorithm and the SFCTSM algorithm when the number of VNFs in the SFC migration request varies among 5, 6, 7 and 8. We first deploy an initial solution to each SFC migration requests, but do not need to allocate resources to them due to these SFC migration requests will be migrated as a whole in the simulation. So each SFC migration request will successfully deploy the initial solution, and in term of the performance in the remapping blocking ratio, the VDC-M algorithm and the SFCTSM algorithm will not affect each other.

From the results, we can see that the remapping blocking ratio of the SFCTSM algorithm is lower than that of the VDC-M algorithm. This is because that the VDC-M algorithm is proposed for the VDC migration, and the SFCTSM algorithm is specifically proposed for the total reconfiguration problem of the SFC migration request in the cloud-fog environment, and the two-step migration strategy of the SFCTSM algorithm can effectively reduce the remapping blocking ratio of the SFC migration request. So the SFCTSM algorithm can obtain a lower remapping blocking ratio than the VDC-M algorithm dose.

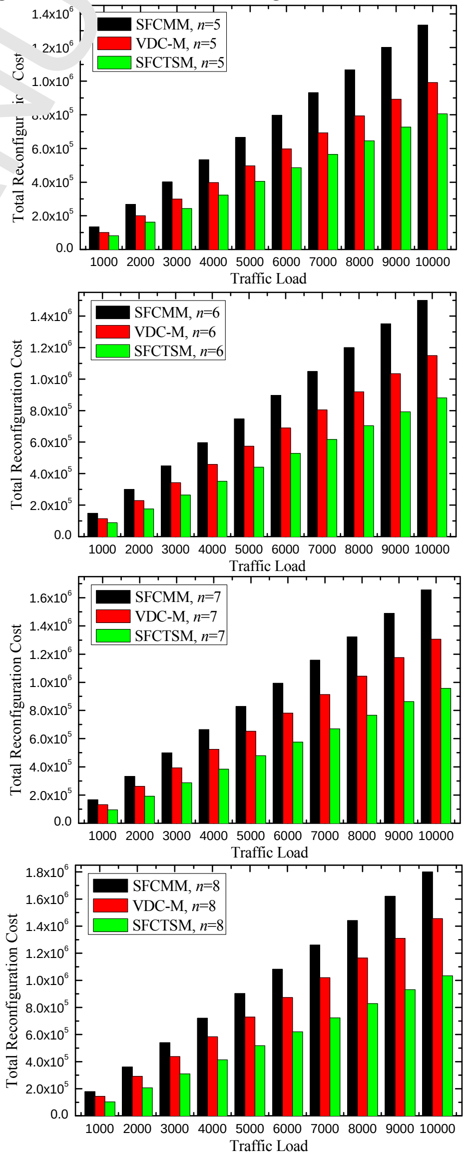


Fig.6 The total reconfiguration cost

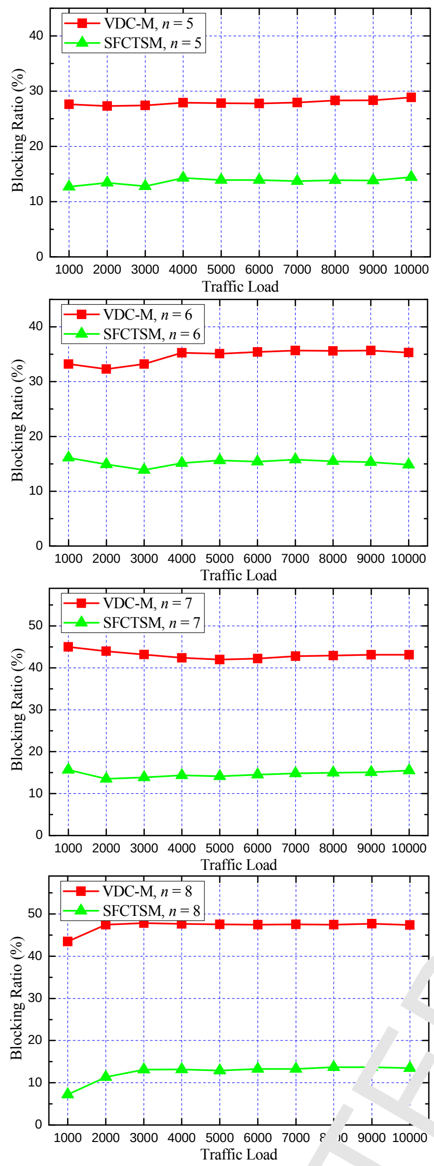


Fig.7 The remapping blocking ratio

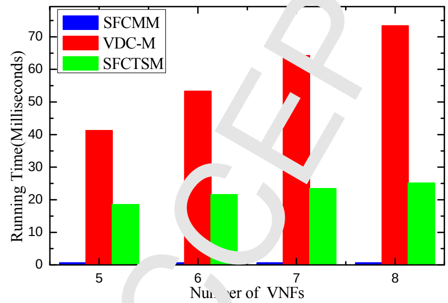


Fig.8 The running time

Fig.8 illustrates the running times of the SFCMM algorithm, VDC-M algorithm and SFCTSM algorithm wherein the number of VNFs in the SFC migration request varies among 5, 6, 7 and 8. From Fig.8, we can see that the running time of the SFCMM algorithm is much lower than that of the VDC-M algorithm and SFCTSM algorithm due to the SFCMM algorithm only remaps the vSGW and the vPGW so that can quickly complete the reconfiguration process of the SFC migration request. Moreover, due to the VDC-M algorithm has more traversal times during execution, the running time of the VDC-M algorithm is higher than that of the SFCTSM algorithm.

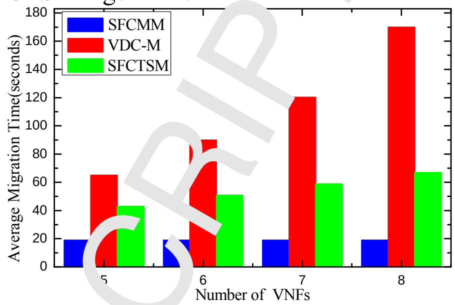


Fig.9 The average migration time

Fig.9 describes the average migration times of the SFC migration request of the SFCMM algorithm, VDC-M algorithm and SFCTSM algorithm wherein the number of VNFs varies among 5, 6, 7 and 8. From results, we can see that due to the SFCMM algorithm only migrates the vSGW and the vPGW, however the VDC-M algorithm and SFCTSM algorithm will migrate all VNFs, the average migration time of the SFC migration request of the SFCMM algorithm is lowest and does not increase with the increase of the number of VNFs. Moreover, the average migration time of the SFC request of the SFCTSM algorithm is lower than that of the VDC-M algorithm due to the SFCTSM algorithm uses the two-step migration strategy to migrate the SFC migration request and the second step migration of the two-step migration strategy uses the post-copy based parallel migration strategy that can reduce the migration time of the SFC migration request.

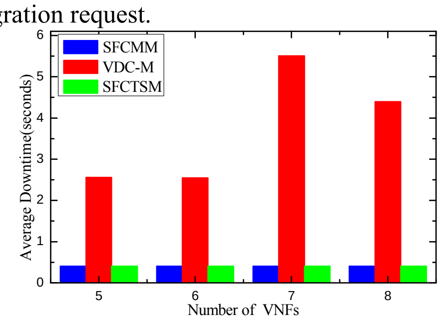


Fig.10 The average downtime

Fig.10 explains the average downtime times of the SFC migration request of the SFCMM algorithm, the VDC-M algorithm and SFCTSM algorithm. Fig.10 shows that the average downtime time of the SFC request of the SFCMM algorithm is lowest and does not increase with the increase of the number of VNFs owing to the SFCMM algorithm only migrates the vSGW and the vPGW that the downtime is also shortest which the shortest migration time is obtained, however the SFCTSM algorithm adopts the two-step migration mechanism for migrating the SFC migration request and only the first step migration in the two-step migration strategy will cause the SFC to shut down and the first step migration is to remap the minimum number of the VNFs, so the average downtime time of the SFC migration request of the SFCTSM algorithm is close to that of the SFCMM algorithm. In addition, the VDC-M algorithm adopts the pre-copy of parallel migration mechanism for migrating all VNFs that will lead to the longest downtime, and since the characteristics of iterative migration, the downtime of the VDC-M algorithm is oscillating up with the number of VNFs increases.

From all simulation results, we can see that the SFCMM algorithm has a shorter migration time and a similar downtime than the SFCTSM algorithm does, however, it will waste the physical network resources that lead to a higher reconfiguration cost. So the SFCMM algorithm applies to quickly migrate and quickly restore the user's service when the service provider does not care to use more physical resources. In contrast, the SFCTSM algorithm has a lower reconfiguration cost, a longer migration time and a similar downtime than the SFCMM algorithm does, so if the service provider is concerned about the utilization of physical resources, the SFCTSM algorithm is more appropriate.

6. Conclusion

In this paper, we study the problem of the SFC migration in the cloud-fog computing environment to optimize the migration performance when the mobile user moves from a fog radio access network to another fog radio access network. To solve the SFC migration problem, we firstly formulate the migration problem of SFCs as the integer linear programming. Then, we propose two SFC migration strategies: the minimum number of VNFs migration strategy and the two-step migration strategy to improve the reconfiguration cost, the migration time, the downtime and the remapping success ratio of the SFC migration request. To optimize the downtime and migration time of the SFC migration request, in the two-step migration strategy, we adopt the pre-copy based parallel migration mechanism for migrating these remapped VNFs in the first step migration, and we adopt the post-copy based parallel migration mechanism for migrating these remapped VNFs in the second step migration. And we numerically analyzed the migration time and the downtime. Then, we have designed a two-step migration algorithm to migrate SFCs based on the two-step migration strategy. Finally, we emulate and evaluate our proposed algorithm in the federated environment of the cloud computing and the fog computing, and simulation results show good performances in the reconfiguration cost, the downtime and migration time and the remapping success ratio of the SFC migration request.

We use the traditional method to solve the SFC migration problem. In our future work, we intend to extend the state of the art taking into consideration also approaches based on computational intelligence and artificial intelligence [46, 47]. The computational intelligence and the artificial intelligence are widely used in various fields and show outstanding performance advantages. I think they will also bring positive benefits to the deployment and migration of SFCs.

Acknowledgement

This research was partially supported by Natural Science Foundation of China (61571098), 111 project (B14039), Fundamental Research Funds for the Central Universities (ZYGX2016J217), the Open Reasearch Foundatoin of Science and Technology on Communication Networks Laboratory under Grant XX17641X011-07.

References

[1] Gang Sun, Dan Liao, Dongcheng Zhao, Zhili Sun, Victor Chang. Towards Provisioning Hybrid Virtual Networks in Federated Cloud Data Centers. Future Generation Computer Systems, 87 (2018) 457-469.

[2] Mosharaf Chowdhury, Muntasir Raihan Rahman, Raouf Boutaba. ViNEYard: Virtual network embedding algorithms with coordinated node and link mapping. IEEE/ACM Transactions on Networking, 1 (20) (2012) 206-219.

[3] Gang Sun, Victor Chang, Guanghua Yang, Dan Liao. The Cost-efficient Deployment of Replica Servers in Virtual Content Distribution Networks for Data Fusion. Information Sciences, 432 (2018) 495-515.

[4] Bing Leng, Liusheng Huang, Chunming Qiao, Hongli Xu. A Light-weight Approach to Obtaining NF State Information in SDN+NFV Networks. IEEE INFOCOM, 2018, pp.1-9.

[5] Marouen Mechtri, Chaima Ghribi, Oussama Soualah, Djamal Zeghlache. NFV Orchestration Framework Addressing SFC Challenges. IEEE Communications Magazine, 55 (6) (2017) 16-23.

[6] Gang Sun, Yayu Li, Hongfang Yu, Athanasios V. Vasilakos, Xiaojiang Du, Mohsen Guizani. Energy-efficient and Traffic-aware Service Function Chaining Orchestration in Multi-Domain Networks. Future Generation Computer Systems, 91 (2019) 347-360.

[7] Gang Sun, Yayu Li, Dan Liao, Victor Chang. Service Function Chain Orchestration across Multiple domains: A Full Mesh Aggregation Approach. IEEE Transactions on Network and Service Management, 15 (3) (2018) 1175-1191.

[8] Linqi Guo, John Pang, Anwar Walid. Joint Placement and Routing of Network Function Chains in Data Centers. IEEE INFOCOM, 2018, pp.1-9.

[9] Wenrui Ma, Oscar Sandoval, Jonathan Beltran, Deng Pan, Niki Pissinou. Traffic Aware Placement of Interdependent NFV Middleboxes. IEEE INFOCOM, 2017, pp.1-9.

[10] Satyam Agarwal, Francesco Malandrino, Carla-Fabiana Chiasserini, Swades De. Joint Placement and Routing of Network Function Chains in Data Centers. IEEE INFOCOM, 2018, pp.1-9.

[11] Gang Sun, Yayu Li, Yao Li, Dan Liao, Victor Chang. Low-Latency Orchestration for Workflow-Oriented Service Function Chain in Edge Computing. Future Generation Computer Systems, 85 (2018) 116-128.

[12] Tung-Wei Kuo, Bang-Heng Liou, Kate Ching-Ju Lin, Ming-Jer Tsai. Deploying Chains of Virtual Network Functions: On the Relation Between Link and Server Usage. IEEE INFOCOM, 2016, pp.1-9.

[13] Zilong Ye, Xiaojun Cao, Jianping Wang, Hongfang Yu, Chunming Qiao. Joint Topology Design and Mapping of Service Function Chains for Efficient, Scalable, and Reliable Network Functions Virtualization. IEEE Network, 30 (3) (2016) 81-87.

[14] Salvatore D'Oro, Laura Galluccio, Sergio Palazzo, Giovanni Schembra. Exploiting Congestion Games to Achieve Distributed Service Chaining in NFV Networks. IEEE Journal on Selected Areas in Communications, 35 (2) (2017) 407-420.

[15] Keke Gai, Meikang Qiu, Hui Zhao, Lixin Tao, Ziliang Zong. Dynamic energy-aware cloudlet-based mobile cloud computing model for green computing. Journal of Network and Computer Applications, 59 (2016) 46-54.

[16] Keke Gai, Meikang Qiu, Hui Zhao. Energy-aware task assignment for mobile cyber-enabled applications in heterogeneous cloud computing. Journal of Parallel and Distributed Computing, 111 (2018) 126-35.

[17] Huaqing Zhang, Yanru Zhang, Yunan Gu, Dusit Niyato, Zhu Han. A Hierarchical Game Framework for Resource Management in Fog Computing. IEEE Communications Magazine, 55 (8) (2017) 52-57.

[18] Zuoxia Yu, Man Ho Au, Qiuliang Xu, Rupeng Yang, Jinguang Han. Towards leakage-resilient fine-grained access control in fog computing. Future Generation Computer Systems, 2017, pp.1-15.

[19] Mung Chiang, Sangtae Ha, Chih-Lin I, Fulvio Risso, Tao Zhang. Clarifying Fog Computing and Networking: 10 Questions and Answers. IEEE Communications Magazine, 55 (4) (2017) 18-20.

[20] Pengfei Hu, Huansheng Ning, Tie Qiu, Yanfei Zhang, Xiong Luo. Fog Computing Based Face Identification and Resolution Scheme in

Internet of Things. IEEE Transactions on Industrial Informatics, 13 (4) (2017) 1910-1920.

[21] Yeonghun Nam, Sooeun Song, Jong-Moon Chung. Clustered NFV Service Chaining Optimization in Mobile Edge Clouds. IEEE Communications Letters, 21 (2) (2017) 350-353.

[22] Frank Van Lingen, Marcelo Yannuzzi, Anuj Jain, Rik Irons-Mclean, Oriol Lluch, David Carrera, Juan Luis Pérez, Alberto Gutierrez, Diego Montero, Josep Martí, Ricard Masó, Juan Pedro Rodríguez. The Unavoidable Convergence of NFV, 5G, and Fog: A Model-Driven Approach to Bridge Cloud and Edge. IEEE Communications Magazine, 55 (8) (2017) 28-35.

[23] Vincenzo Eramo, Emanuele Miucci, Mostafa Ammar, Francesco Giacinto Lavacca. An Approach for Service Function Chain Routing and Virtual Function Network Instance Migration in Network Function Virtualization Architectures. IEEE/ACM Transactions on Networking, 25 (4) (2017) 2008-2025.

[24] Vincenzo Eramo, Mostafa Ammar, Francesco Giacinto Lavacca. Migration Energy Aware Reconfigurations of Virtual Network Function Instances in NFV Architectures. IEEE Access, 5 (2017) 4927-4938.

[25] Francisco Carpio, Admela Jukan, Rastin Pries. Balancing the Migration of Virtual Network Functions with Replications in Data Centers. IEEE/IFIP Network Operations and Management Symposium: Cognitive Management in a Cyber World, 2018, pp.1-8.

[26] Gang Sun, Dan Liao, Dongcheng Zhao, Zichuan Xu, Hongfang Yu. Live Migration for Multiple Correlated Virtual Machines in Cloud-based Data Centers. IEEE Transactions on Services Computing, 11 (2) (2018) 279-291.

[27] Yu-Jen Ku, Dian-Yu Lin, Chia-Fu Lee, Ping-Jung Hsieh, Hung-Yu Wei, Chun-Ting Chou, Ai-Chun Pang. 5G Radio Access Network Design with the Fog Paradigm: Confluence of Communications and Computing. IEEE Communications Magazine, 55 (4) (2017) 46-52.

[28] Nicola Iotti, Marco Picone, Simone Cirani, Gianluigi Ferrari. Improving Quality of Experience in Future Wireless Access Networks through Fog Computing. IEEE Internet Computing, 21 (2) (2017) 26-33.

[29] Kai Liang, Liqiang Zhao, Xiaoli Chu, Hsiao-Hwa Chen. An Integrated Architecture for Software Defned and Virtualized Radio Access Networks with Fog Computing. IEEE Network, 31 (1) (2017) 80-87.

[30] Seongjin Park, Younghwan Yoo. Network Intelligence Based on Network State Information for Connected Vehicles Utilizing Fog Computing. Mobile Information Systems, 43 (12) (2017) 1420-1427.

[31] Mehdi Sookhak, F. Richard Yu, Ying He, Hamid Talebian, Nader Sohrabi Safa, Nan Zhao, Muhammad Khurram Khan, Neeraj Kumar. Fog Vehicular Computing: Augmentation of Fog Computing Using Vehicular Cloud Computing. IEEE Vehicular Technology Magazine, 12 (3) (2017) 55-64.

[32] Quanying Sun, Ping Lu, Wei Lu, Zuqing Zhu. Forecast-Assisted NFV Service Chain Deployment based on Affiliation-Aware vNF Placement. IEEE Globecom, 2016, pp.1-6.

[33] Zichuan Xu, Weifa Liang, Alex Galis, Yu Ma. Throughput Maximization and Resource Optimization in NFV-Enabled Networks. IEEE ICC, 2017, pp.1-7.

[34] Ke Yang, Hong Zhang, Peilin Hong. Energy-aware Service Function Placement for Service Function Chaining in Data Centers. IEEE Globecom, 2016, pp.1-6.

[35] Chuan Pham, Nguyen H. Tran, Shaolei Ren, Walid Saad, Choong Seon Hong. Traffic-aware and Energy-efficient vNF Placement for Service Chaining: Joint Sampling and Matching Approach. IEEE Transactions on Services Computing, DOI:10.1109/TSC.2017.2671867.

[36] Jiajia Liu, Zhongyuan Jiang, Nei Kato, Osamu Akashi, Atsushi Takahara. Reliability Evaluation for NFV Deployment of Future Mobile Broadband Networks. IEEE Wireless Communications, 23 (3) (2016) 90-96.

[37] Alireza Shameli-Sendi, Yosr Jarraya, Makan Pourzandi, Mohamed Cheriet. Efficient Provisioning of Security Service Function Chaining Using Network Security Defense Patterns. IEEE Transactions on Services Computing, DOI:10.1109/TSC.2016.2616867.

[38] Nicolas Herbaut, Daniel Negru, David Dietrich, Panagiotis Papadimitriou. Service Chain Modeling and Embedding for NFV-based Content Delivery. IEEE ICC, 2017, pp.1-6.

[39] Ori Rottenstreich, Isaac Keslassy, Yoram Revah, Aviran Kadosh. Minimizing Delay in Network Function Virtualization with Shared Pipelines. IEEE Transactions on Parallel and Distributed Systems, 28 (1) (2017) 156-169.

[40] Leonhard Nobach, Ivica Rimac, Volker Hilt, David Hausheer. SliM: Enabling Efficient, Seamless NFV State Migration. IEEE 24th International Conference on Network Protocols (ICNP), 2016, pp.1-2.

[41] Akira Misawa, Konomi Mochizuki, Hideo Tsuchiya, Masahiro Nakagawa, Kyota Hattori, Masaru Katayama, Jun-ichi Kani. Proposal on virtual edge architecture using virtual network function live migration with wavelength ADM. 21st Asia-Pacific Conference on Communications (APCC), 2015, pp.327-331.

[42] Jing Xia, Deming Pang, Zhiping Cai, Ming Xu, Gang Hu. Reasonably Migrating Virtual Machine in NFV-featured Networks. IEEE International Conference on Computer and Information Technology (CIT), 2016, pp.361-366.

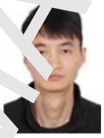
[43] Jing Xia, Zhiping Cai, Ming Xu. Optimized Virtual Network Functions Migration for NFV. IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS), pp.340-346, 2016.

[44] Walter Cerroni, Franco Callegati. Live Migration of Virtual Network Functions in Cloud-Based Edge Network. IEEE ICC, 2014, pp.2963-2968.

[45] Gang Sun, Dan Liao, Vishal Anand, Dongcheng Zhao, Hongfang Yu. A New Technique for Efficient Live Migration of Multiple Virtual Machines. Future Generation Computer Systems, 55 (2016) 74-86,.

[46] Keke Gai, Meikang Qiu. Reinforcement Learning-based Content-Centric Services in Mobile Sensing. IEEE Network, 32 (4) (2018) 34-39

[47] Dan Liao, Yulong Wu, Ziyang Wu, Zeyuan Zhu, Wanting Zhang, Gang Sun, Victor Chang.AI-based software-defined virtual network function scheduling with delay optimization. Cluster Computing, DOI: 10.1007/s10586-018-2124-0.



Dongcheng Zhao is pursuing his Ph.D. degree in Communication and Information System at University of Electronic Science and Technology of China. His research interests include network function virtualization, cloud computing, fog computing and 5G mobile networks.

Dan Liao is a professor at University of Electronic Science and Technology of China (UESTC). He received his B.S. degree in Electrical Engineering in 2001 from UESTC, and his Ph.D. degree in Communication and Information Engineering in 2007 from University of Electronic Science and Technology of China, respectively. His research interests are in the area of wired and wireless computer communication networks and protocols, next generation network.

Gang Sun is an associate professor at University of Electronic Science and Technology of China (UESTC). He received his Ph.D. degree in Communication and Information Engineering in 2012 from University of Electronic Science and Technology of China. His research interests are in the area of network security, datacenter networking and cloud computing.

Shizhong Xu received the B.S., M.S., and Ph.D. degrees in electrical engineering from University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 1994, 1997 and 2000, respectively. He is now a Professor in UESTC. His research interests include broadband networks, all-optical network, and next-generation networks.

Victor Chang is an associate professor at Xi'an Jiaotong Liverpool University, Suzhou, China, after working as a Senior Lecturer at Leeds Beckett University, UK, for 3.5 years. Within 4 years, he completed PhD (CS, Southampton) in 2013 and PGCert (Higher Education, Fellow) in 2012 part-time. He won a European Award on Cloud Migration in 2011 and best papers in 2012 and 2015, and numerous awards since 2012. He is a leading expert on Big Data/Cloud/security, visiting scholar/PhD examiner at several universities, an Editor-in-Chief of IJOCI & OJBD journals, Editor of FGCS, founding chair of two international workshops and founding Conference Chair of IoTBD 2016 www.iotbd.org and COMPLEXIS 2016 www.complexis.org.