Redundancy mechanism of Service Function Chain with Node-Ranking Algorithm

Leyi Zhang, Ying Wang, Xuesong Qiu, Hantao Guo

State Key Laboratory of Networking and Switching Technology

Beijing University of Posts and Telecommunications

Beijing, China

Email: {ly_z, wangyg }@bupt.edu.cn

Abstract*—Network Function Virtualization (NFV) decouples network functions from dedicated hardware platforms, and reduces the cost of proprietary physical equipment. In NFV environment, Service Function Chain (SFC) combines multiple network functions to deal with traffic, and makes the deployment of network services more flexible. However, this new form of end-to end service brings reliability challenges. Thus, it is essential to avoid service interruption in the event of failure. By using Node-ranking algorithm with Centrality and Reliability (NRCR), we design a redundancy mechanism that considers the sharing of functions and the reliability of physical devices to protect the service from interruption. The simulation results show that our method can reduce the backup consumption by up to 39% with respect to the present algorithms and get a higher SFC requests acceptance ratio.

***Keywords*—NFV, SFC, redundancy, reliability, centrality**

I. INTRODUCTION

Under the background of the rapid development of the network, network operators become concerned with providing more convenient, faster and more efficient network services to customers. Traditionally, network operators often steer traffic flow through a list of middleboxes according to a predefined order to accomplish the network service[1]. However, middleboxes are built on specialized hardware and offer single-purpose functionality, leading to high capital expenditures (CAPEX) and operational expenditures (OPEX)[2]. Fortunately, NFV allows for the consolidation of many network equipment types onto high volume servers, switches and storage, which could be located in data centers, distributed network nodes and at end user premises.[3] To deal with a wide range of services, operators apply SFC which consists of a set of ordered VNFs[4]. Although it brings flexibility and extensibility, the potential causes of VNF failure are more complex which are different from traditional proprietary equipment[5].

There are two main methods to deal with failures. One is remapping after offering service and the other is redundancy before the network runs. Most researches adopt redundancy method in the NFV environment while remapping is more suitable for virtual network embedding (VNE). In SFC redundancy, Fan et al. [6] start earlier researching reliable SFC mapping in this field. They think that typical 1:1 redundant architecture has proven to be ineffective. In their works, they proposed a novel enhanced Joint Protection (JP) to share the backup instance. While it cannot solve mapping problem in a polynomial time, they use an online algorithm called GREP[7] to achieve local optimum. GREP, in the selection phase, tends to choose the most unreliable VNF in each chain. By this way, although backup instances can quickly improve the reliability of SFCs, it ignores the share of VNF instances and reliability of backup instance that results in low cost-efficiency. Ding's research[8] tends to start from the overall forwarding graph and consider the sharing of VNF instances. VNFs that are shared by multiple chains have a higher priority in backup. This strategy can reduce backup costs to a certain extent. In VNE situation, In case of link failure, for example, SVNE[9], originally proposed by Muntasir, has removed the assumption that the infrastructure provider (InP) network remains operational at all times, and develops a hybrid policy heuristic to solve the problem. They propose a profit driven method to overcome single substrate link failure. In the case of node failure, the topology of VN is an important factor. In order to ensure the recovery of important nodes from single substrate node failure, the authors of [10] design a redundant VN topology by adding nodes and links to the original VN.

However, even if redundancy can protect SFC from failures, we still hold several problems to solve urgently. Therefore, in this paper, we investigate which VNF we should backup and on which physical position we should deploy the VNF instance. Inspired by previous studies, we address the problem of reliable SFC backup and propose an offline algorithm called NRCR which uses a novel selection model to choose VNF in forwarding graph after mapping SFCs to substrate network and a cost-efficient backup deployment mechanism to cut down the backup cost under the premise of meeting the reliability requirement. In summary, we list the main contributions as follows:

We consider the VNF's centrality in forwarding graph as an important indicator to evaluate VNFs, which consists of the degree of VNF instances and their bandwidth requirements.

We propose a Node-Ranking algorithm that consider both the centrality and reliability of VNF to choose the most worthwhile VNF to be backed up. Compared with other solutions, this method that integrates two phases reduces the cost of backup by 30%.

The rest of this paper is organized as follows. Section II discusses the related work. Section III describes the problem statement and formulation. Section IV introduces the proposed NRCR algorithm. Simulation results are presented in Section V and this paper is concluded in Section VI.

978-3-903176-15-7

II. PROBLEM STATEMENT

In this section, we firstly describe substrate network and VNF-forwarding graph (VNF-FG) [11]. Then we introduce reliability and present the reliability of VNF and SFC. At last, we define the backup problem and give the relevant constraints and objective function.

A. Substrate Network and VNF-FG

Substrate Network is given as $G_s(N_s, L_s)$, which is an undirected graph. N_s represents a set of physical nodes, and nodes can be defined by $PN(C_n, T_n, R_n)$. Let c_n be the node capability, t_n the set of VNF types that node can carry, and r_n as the reliability of each physical node. L_s represents the physical link between nodes.

VNF-FG is formed by the mapping of primary multiple service function chains. We first define the structure of the SFC. The structure is described as $Req_i(F_i, R_i, P_i)$. F_i is the set of VNFs required by the service request. The VNF can be represented by (c_j, b_j, t_j). c_j and b_j respectively define its node resource and outgoing bandwidth requirements. t_j defines the function type. Here we assume that the same function type will not be required in one request. R_i is the minimum requirement of reliability. P_i is the location of the physical node to which each function is specifically mapped. As we only discuss the backup process in this paper, we assume that SFCs have been mapped to physical network to get an initial mapping. The structure of VNF-FG is $G_v(F_v, L_v)$, F_v represents all function instances in the graph, and L_v represents the virtual link connection between the instances.

B. Reliability

As the probability of instances failure is relatively small and it is more complex to consider two situations together, so in our paper, we only consider the service break caused by physical equipment failures. The reliability of the physical node r_n can be estimated using the *Mean Time Between Failure* (MTBF). When a VNF instance is mapped to a physical node, the reliability of the instance is equal to the reliability of the physical node. Assuming that SFC can provide services only when all required features are working properly, each SFC's reliability is defined as follows:

$$r_i = \prod_{f \in F_i} r_f \qquad (1)$$

C. Problem Definition

We set up the issue in an offline scenario. After constructing SFCs according to the service request from the user, the SFCs are mapped to the physical network in turn. The mapping of primary SFCs are often unsatisfactory, for the overall reliability of chains cannot meet the user's requirement. Therefore, backup operations need to be performed before the network is officially operational. To minimize the consumption of resources, we define the backup problem in the paper as: *Given a number of already mapped SFCs, namely VNF-FG, we need to give an optimal backup scheme to upgrade SFCs' reliability which doesn't reach the demand and optimize link bandwidth consumption.* Then we will use two decision variables to indicate the input and output of our problem.

$x_n^v \in \{0,1\}$ -indicates whether VNF instance v is mapped to physical node n.

$x_n^b \in \{0,1\}$ -indicates whether backup instance b is mapped to physical node n.

Based on the above input and output, we now present the objective function and its constraints. The mathematical formulation is

Objective:

$$\min \sum_{j \in F_i} b_j \qquad (2)$$

Subject to:

$$r_i' \geq R_i \text{ for } i \in I \qquad (3)$$

$$x_n^v + x_n^b \leq 1 \text{ for } v, b \in F_v, n \in N \qquad (4)$$

$$t_b \in t_n \text{ for } b \in F_v, n \in N \qquad (5)$$

$$\sum_{b \in F_v} x_n^b c_j \leq c_n \text{ for } j \in F_i, n \in N \qquad (6)$$

The objective function (2) aims at minimizing the bandwidth consumption every time a backup instance is deployed. Constraint (3) ensures that the reliability for each service request after backup should exceed the user's demand. Meanwhile, constraint (4) ensures that backup instance and primary instance should not be mapped on the same physical node. Constraint (5) ensures that backup instance should be mapped to the node which can carry its function type. Last, constraint (6) ensures that the sum of resource requirement of backup instance should not exceed the capacity of physical node.

III. ALGORITHM DESIGN AND IMPLEMENTATION

In this section, we propose a novel Node-Ranking algorithm to select VNF for backup. In order to achieve the goal of optimizing the backup link consumption, we use the BFS algorithm in the process of selecting physical nodes.

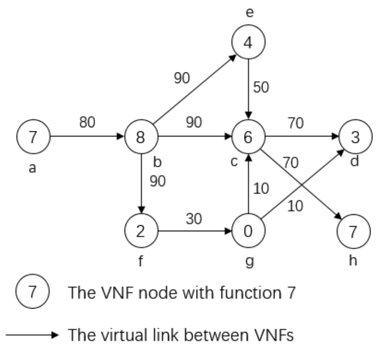

Fig. 1.A VNF-FG with 8 nodes and 9 weighted links.

A. Centrality

In order to select the appropriate VNF for backup, we consider the various topological characteristics of the VNF in the forwarding graph. Here we introduce the centrality of graph theory. The centrality of a node consists of its degree and link weight. Since the centrality of a node shows the relationship between the node and its adjacent nodes, we explain why we should find the VNF node with the highest centrality to back up below.

For instance, Fig.1 shows a VNF-FG in which the number in the node represents the type of VNF required by the current node and the number above lines represents the outgoing bandwidth. The degree of nodes *b* and *c* in the graph is higher(in this case, the in-degree and out-degree are not distinguished), which means that there are probably more SFCs sharing this node. Backing up such nodes can improve the reliability of multiple SFCs in a single backup process and is a very effective strategy.

However, if we only use the sharing factor, the overhead of backup may be too large because more bandwidth resources will be used when establishing backup links for backup node with higher degree. Therefore, considering both the degree of the node itself(request sharing) and the link weight (bandwidth overhead) between nodes can optimally solve the current problem. Thus, we introduce the centrality of nodes into the forwarding graph, the centrality of each node is defined as:

$$C_{in}(i,j) = d_{in}^{1-\alpha} \cdot b_{in}^{\alpha} \quad (7)$$

$$C_{out}(i,j) = d_{out}^{1-\alpha} \cdot b_{out}^{\alpha} \quad (8)$$

$$C(i,j) = C_{in}(i,j) + C_{out}(i,j) \quad (9)$$

The in-degree of a node can be quantified by the number of links that originate from a node, d_{in} while the out-degree of is the number of links that are directed towards a node, d_{out}. Relatively, for this weighted network, b_{in} and b_{out} can be defined as the total bandwidth requests attached to the outgoing and incoming links. In Eq.(7) and Eq.(8), we respectively formulate the in-centrality and out-centrality of a VNF node. In an attempt to combine both degree and bandwidth requests, we use a tuning parameter α in the Eq.(7) and Eq.(8), which determines the relative importance of the number of links compared to weights. More specifically, α is a positive parameter that can be set according to the real data. And in this paper, we use the sum of influences of the outgoing and incoming links as the centrality of a VNF as in Eq.(9).

B. Backup Selection

It's not enough to prove that the VNF node with the highest centrality is the most worthwhile instance to be backed up. We need to incorporate the reliability of the VNF into the backup selection process. When the VNF is initially mapped, each instance get an primary reliability from the physical device. When we choose the VNF node to back up, we should consider both the centrality and the reliability for improving the centrality mainly utilizes the topology characteristics of the forwarding graph to improve the efficiency of backup, and improving the reliability can quickly meet the user's demand for services. Combining two factors above to make an evaluation value for each VNF node, we can comprehensively assess which node is the most worthwhile to back up. Obviously, backup process doesn't only select the VNF instance once. In order to ensure that we can find the instance with the highest evaluation value every time, here we use the Node-Ranking algorithm to determine the VNF which needs to be backed up. The evaluation value that a VNF instance uses for ranking is defined as:

$$V(i,j) = C(i,j) \cdot \cos\left(\frac{\pi}{2} \cdot \frac{r_{ij} - r_{\min}}{r_{\max} - r_{\min}}\right) \quad (10)$$

In Eq.(10), a trigonometric function is used to control the effect of reliability on the evaluation value of centrality and reliability (VCR). After the selected VNF is backed up, the VCR of the VNF should be significantly decreased. In this way, different VNFs are selected in each backup process, which indirectly solves the problem of the number of backups. $r_{\min}$ refers to the minimum reliability of the service request requested by the user, and $r_{\max}$ is the highest reliability required. The purpose of setting two values is to prevent the algorithm from falling into an infinite loop. If we backup a SFC with the highest VCR for several times, it gets little reliability increment, thus we set the VNF which achieves the maximum reliability to negative value though this kind of setting and choose other VNFs to back up. This method increasing the acceptance rate in multiple SFC situations.

After defining the VCR with centrality and reliability of VNF, we now discuss the details of NRCR algorithm. The pseudo-code of the whole redundancy process is presented in Alg.1. The algorithm is a kind of greedy algorithm. The input are the underlying physical topology, VNF-FG and the initial mapping scheme (line 1), and we perform backup processing for each SFC that does not meet the reliability requirements (line 4). In the loop process, we always select the VNF with the highest VCR (line 6), and then select the appropriate physical node to deploy the backup instance (line 10), and finally outputs the optimized backup plan.

Algorithm 1 Node-Ranking Algorithm

1: Input: $G_s(N_s, L_s)$, $G_v(F_v, L_v)$, $Q = \{Req_i | i \in I\}$,
   The primary placement scheme $\{X_n^v\}$
2: Output: the backup plan $\{X_n^b\}$
3: **for** each $\mathrm{Re}\,q_i, i \in I$ **do**
4: **while** $r_i < R_i$ **do**
5: **for** v *in* r_i **do**
6: $v_i \leftarrow$ Select the VNF with maximum VCR
7: **end for**
8: **if** $v_j \in V$ **then**
9: $b_j = v_j$
10: $n_j \leftarrow$ Select the physical node using BFS
11: Update r_{ij}, r_i
12: **end if**
13: **end while**
14: **end for**
15: return $\{X_n^v\}$

IV. SIMULATION

All simulations are implemented on a server with Intel core 4, 3.3 GHz and 8GB RAM. For physical network, we use a network topology with 27-nodes[12]. Moreover, the reliability of these nodes is set as a random value distributed between 0.9 and 0.999. In order to realize multiple SFCs deployment, each PN node can implement three VNF types (the value of type is also get randomly) but for simplicity a node can only implement one kind of VNF type if it has deployed a instance whether the instance is primary or backup. For the service function chains, each SFC request consists of two to five VNFs in series. In the light of the SLA requirement of Google Apps [13], the reliability requirement of each SFC requests is selected among [0.95, 0.98, 0.99, 0.995, 0.999]. In the VNF-FG, we simulate different number of primary VNF instances that are placed in the physical network to implement up to multiple SFC requests. These requests share several primary instances and the results of our simulation are average value of different VNF-FG. We compared the NRCR scheme with the other two algorithms, MaxR and MinC. The target of MaxR is to get the highest reliability increment while the target of MinC is to get the least node resource consumption.

From Fig.2, NRCR outperforms MaxR and MinR by up to 39% and 21% backup consumption reduction. MaxR consumes a lot of bandwidth in the process of pursuing the reliability increment. Since it requires lower reliability requirements, the consumption of NRCR is much less than that of MinC. Next, we compare NRCR with GREP. From Fig.3, the service request acceptance ratio of the NRCR is about 93%. GREP usually accepts only 80%.

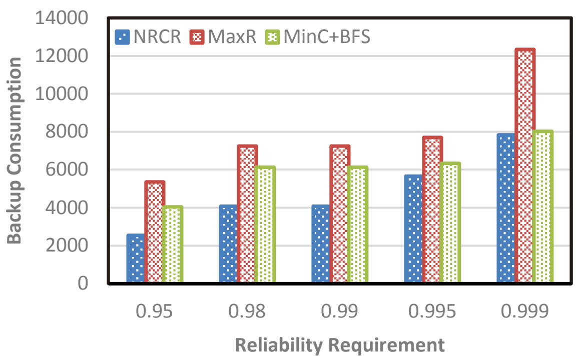

Fig. 2.Backup Consumption

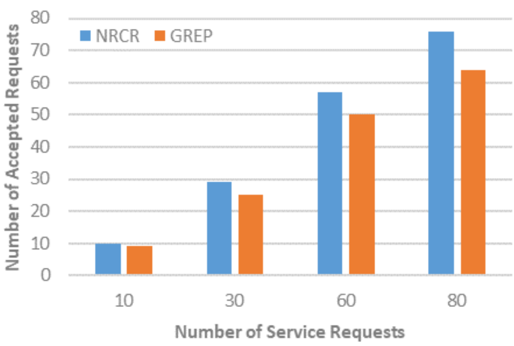

Fig. 3.SFCs acceptance ratio

We at last discuss its time complexity here. The selection process traverses all service requests and every SFC is backed up in the worst case. In the selection algorithm of the physical node, the BFS mode only requires the complexity of $O(N+e)$ (e is the number of undirected edges of the physical network), but if the VNF to be backed up is an intermediate node, the complexity is $O(N^3)$.

V. CONCLUSION

SFC in NFV can realize and offer services automatically and swiftly. However, this new service deployment also brings inevitable reliability problem. In this paper, we make full use of the characteristics of function sharing, use the centrality principle and reliability to evaluate the network function in the service request. Then we propose a heuristic algorithm called NRCR to solve the problem of service backup. The simulation results show our approach can reduce backup consumption by about 39% and increase the acceptance ratio of multi-service requests.

ACKNOWLEGEMENT

This work was supported by the National Natural Science Foundation of China (61501044).

REFERENCE

[1] P. Quinn and T. Nadeau, "Service function chaining problem statement," 2013.

[2] J. Soares and S. Sargento, "Optimizing the embedding of virtualized cloud network infrastructures across multiple domains," in IEEE International Conference on Communications, 2015, pp. 442–447.

[3] R. Mijumbi, J. Serrat, J. L. Gorricho, N. Bouten, F. D. Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," IEEE Communications Surveys & Tutorials, vol. 18,no. 1, pp. 236–262, 2015.

[4] J. Halpern E, C. Pignataro E. Service Function Chaining (SFC) Architecture[J]. 2015.

[5] D. Cotroneo, L. De Simone, A. Iannillo, A. Lanzaro, R. Natella, J. Fan, and W. Ping, "Network function virtualization: challenges and directions for reliability assurance," in Software Reliability Engineering Workshops (ISSREW), 2014 IEEE International Symposium on. IEEE, 2014, pp. 37–42.

[6] Fan J, Guan C, Zhao Y, et al. Availability-aware mapping of service function chains[C]// IEEE INFOCOM 2017 - IEEE Conference on Computer Communications. IEEE, 2017:1-9.

[7] Fan J, Ye Z, Guan C, et al. GREP:Guaranteeing Reliability with Enhanced Protection in NFV[C]// ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization. ACM, 2015:13-18.

[8] Ding W, Yu H, Luo S. Enhancing the reliability of services in NFV with the cost-efficient redundancy scheme[C]// IEEE International Conference on Communications. IEEE, 2017:1-6.

[9] Rahman M R, Boutaba R. SVNE: Survivable Virtual Network Embedding Algorithms for Network Virtualization[J]. IEEE Transactions on Network & Service Management, 2013, 10(2):105-118.

[10] Yu H, An and V , Qiao C and Sun G , "Cost Efficient Design of Survivable Virtual Infrastructure to Recover from Facility Node Failures," in *Proc. of IEEE International Conference on Communications (ICC)*, pp. 1-6, June 5-9, 2011.

[11] "ETSI GS NFV 001 V1.1.1: Network Function Virtualization Use Cases," http://www.etsi.org/deliver/etsi gs/NFVINF/ 001 099/001/01.01.01 60/gs NFVINF001v010101p.pdf, ETSI ISG NFV, Tech. Rep., 2013.

[12] Orlowski S , Wess?Ly R , M. Pióro, et al. SNDlib 1.0—Survivable Network Design Library[J]. Networks, 2010, 55(3):0-0.

[13] "Google Apps Service Level Agreement." [Online]. Available: http://www.google.com/apps/intl/en/terms/sla.html