

Poster Abstract: Joint Optimization of Service Function Chain Elastic Scaling and Routing

Weihan Chen^{*}, Zhiliang Wang^{*}, Han Zhang[†], Xia Yin^{*}, Xingang Shi^{*}, Lei Sun[‡]

^{*}Tsinghua University, China

[†]Beihang University, China

[‡]Lenovo, China

chenwh18@mails.tsinghua.edu.cn, wzl@cernet.edu.cn

Abstract—The main problem of current Virtualization Network Function (VNF) elastic scaling mechanism is that it does not consider the routing cost variation of the whole Service Function Chain (SFC) after scaling operation. In some network environment, the routing cost may increase considerably. In this poster, we address the joint optimization problem of SFC elastic scaling and routing, and propose a greedy algorithm to reduce routing cost caused by scaling operation. The main idea is to force the scaling of VNF, which does not require scaling operations, to optimize the routing cost of traffic forwarding paths. The simulation results indicate that the proposed algorithm can reduce 27% total cost compared with traditional scaling algorithm.

I. INTRODUCTION

For network services providers, one of the most important challenges in Network Function Virtualization (NFV) based environment is to increase economic revenue by efficiently allocate computing resources to dynamic user demands. NFV supports corresponding scaling mechanisms to adapt to dynamics of network traffic. There are three main types of scaling mechanisms: 1) Horizontal scaling (scale out/in): add or remove VNF instances (VNFI); 2) Vertical scaling (scale up/down): increase or decrease computing resource (CPU, RAM, etc.) of VNFI; 3) VNF migration. Meanwhile, scaling algorithms are also required to minimize the cost caused by scaling operation while meeting user demands. The cost mainly includes traffic routing cost (e.g. latency) and VNF scaling cost, where VNF scaling cost is consist of computing resource cost (e.g. computing resource maintenance expenditure) and service interruption cost (VNF scaling may cause service stop). Reducing VNF scaling cost means reducing maintenance expenditure and losses from service interruption, and reducing routing cost means providing higher quality services (high revenue service), which can increase economic benefits.

Recent researches [1]–[3] mainly focus on the VNFI creation/destruction cost and computing (or bandwidth) resource increment/reduction cost, the routing cost is usually ignored. When horizontal scaling is applied, the virtual network function forwarding graph of SFC will change, which results in routing cost variation. In some network environments (like Data Center), the routing cost is less affected, hence it is possible to ignore routing cost. However, in some network environments where the routing cost is greatly affected by

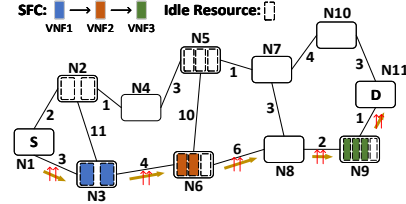


Fig. 1. Scale Out Operation Cost Example

routing variation (like Wide-Area Network), the difference in routing costs may be large and cannot be easily ignored. It is necessary to optimize both routing cost and VNF scaling cost during scaling operation. In this poster, we mainly address the joint optimization problem of SFC scaling and routing (which is NP-Hard) in the case of horizontal scaling, and then develop a possible greedy algorithm to solve the problem.

II. MOTIVATION

In NFV based environment, each physical node has some computing resources to run VNF, and the routing costs (i.e., traffic transmission delay, can be abstracted as weight value) of the paths are also different. When user traffic through the SFC changes, it may cause the resources utilization of VNFs overloading or underloading, and result in service performance degradation or unnecessary resource overhead. Hence, dynamic adjustment of resources allocation of VNFs in the SFC is necessary.

Here, we use a scaling out example to explain the cost caused by scaling operation. As shown in Fig. 1, the SFC includes three VNFs, which are initially deployed on the physical nodes $N3$, $N6$, and $N9$, respectively. The source and destination nodes of user traffic are $N1$ and $N11$. When user traffic increases, $VNF1$ must be scaled out due to insufficient computing resources on $N3$, $VNF2$ can be scaled out on other physical nodes or scaled up directly on $N6$ since existing computing resources is insufficient and there is idle resources on $N6$, and $VNF3$ does not need scaling out/up since it has enough computing resource. The total cost caused by traffic increasing includes additional computing resources cost for scaling $VNF1$ and $VNF2$, service interruption cost caused by scaling operation, and additional routing cost for increasing traffic.

If traditional scaling algorithms are adopted (without routing optimization), $VNF1$ will be scaled out on $N2$, and $VNF2$

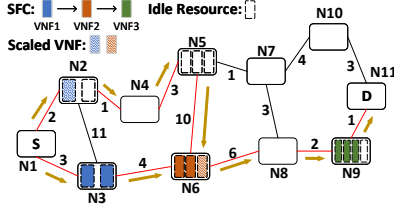


Fig. 2. Example of Total Cost of Scaling Out Without Routing Optimization

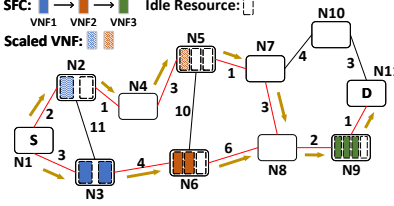


Fig. 3. Example of Total Cost of Scaling Out With Routing Optimization

will be scaled up (since scaling up consumes less cost than scaling out). The routing paths for increasing traffic are shown in Fig. 2. The total routing cost depends on the path with the maximal sum of weight value, $C_{ro} = 25$. Suppose additional computing resource cost for scaling out and up VNFs are abstracted as integer values 2 and 1, respectively. The total resource cost will be the sum of all required computing resource cost, expressed by C_{res} (In this situation, $C_{res} = 3$). The service interruption cost mainly depends on the interruption time, since scaling out takes longer time than scaling up, the service interruption cost is assumed to be the time required for scaling out operation, which is abstracted as an integer value $C_{time} = 3$. The total cost introduced by scaling out will be the weighted sum of routing cost, additional computing resources cost and service interruption cost, $C_{total} = \alpha C_{ro} + \beta C_{res} + \gamma C_{time}$. Weight factors α, β, γ are used to balance these costs, since they are in different dimensions. Assume $\alpha = 1.8, \beta = 1.5, \gamma = 1.3$, the total cost when traditional scaling cost adopted will be: $C_{total} = 53.4$.

In fact, there are some paths with lower routing cost can be used to forward user increasing traffic. If we scale out $VNF2$ on $N5$ instead of scaling up, the traffic forwarding path will change (As shown in Fig. 3). Now, the total routing cost and total computing resource cost will change to $C_{ro} = 16$ and $C_{res} = 4$ respectively. And the service interruption cost will not change. Hence, the total cost with routing optimization will be $C_{total} = 38.7$, which is smaller than the cost without routing optimization. According to this example, we can see that there is optimization space in SFC elastic scaling total cost by combining routing optimization.

The main idea of routing optimization is: force to scale the VNFs which may not require scaling on the paths with lower routing cost. If the saving routing cost is greater than the additional VNF scaling cost, the total cost (includes routing cost and VNF scaling cost) will be smaller. The main objective of traditional scaling algorithms is to reduce VNF scaling related expenses. But if we further optimize the routing, we can provide better service to users, which can also

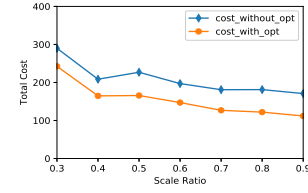


Fig. 4. Total Cost Under Different Scale Ratio

generate additional economic benefits. Routing optimization makes sense when the economic benefits bring by routing optimization outweigh the costs introduced in the process.

III. ALGORITHM

In routing optimization process, we need to tentatively scale each VNF which may not require scaling operation during traversing SFC. And determining how a VNF can be scaled with a minimum cost can be considered a problem similar to static version of EVNFP [1], which can be generalized to the NP-Hard location routing problem. Hence, our problem is also NP-hard. We develop a greedy algorithm to solve it.

The algorithm can be divided into scaling out and scaling in parts. The main idea of scaling out part is: Traversing known VNFs needed to be scaled out (obtained by traditional scaling algorithm), find the middle VNFs and top K shortest paths between every two adjacent VNFs. Then select the path which has the minimum cost and can accommodate the middle VNFs need to be scaled. As for scale in part, the main idea is: Traversing VNFs of SFC, scale in the VNF if it is possible. Finally, re-calculate the total cost after optimization. If it is smaller than the cost without optimization, update the SFC.

IV. RESULT AND FUTURE WORK

We use the backbone network topology CERNET (41 nodes, 59 links) from TopologyZoo [4]. The length of SFC in the topology is 4. And the traffic is generated by Poisson process. The resource capacity and link weight are generated randomly. We calculate the total cost before and after using the optimization algorithm under different scale event ratios respectively. The simulation results are shown in Fig. 4. According to the result, our optimization algorithm can effectively reduce the overall scale cost (27% for CERNET network). In the future, we will research more accurate mathematical model and efficient algorithm.

REFERENCES

- [1] M. Ghaznavi, A. Khan, N. Shahriar, K. Alsubhi, R. Ahmed, and R. Boutaba, "Elastic virtual network function placement," in *2015 IEEE 4th International Conference on Cloud Networking (CloudNet)*, IEEE, 2015, pp. 255–260.
- [2] O. Houidi, O. Soualah, W. Louati, M. Mechtri, D. Zeghlache, and F. Kamoun, "An efficient algorithm for virtual network function scaling," in *GLOBECOM 2017-2017 IEEE Global Communications Conference*, IEEE, 2017, pp. 1–7.
- [3] X. Fei, F. Liu, H. Xu, and H. Jin, "Adaptive vnf scaling and flow routing with proactive demand prediction," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, IEEE, 2018, pp. 486–494.
- [4] S. Knight, H. X. Nguyen, N. Falkner, R. Bowden, and M. Roughan, "The internet topology zoo," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 9, pp. 1765–1775, 2011.