Undergraduate Research Abstract:

Efficient Multicasting Algorithm in SDN Networks Supporting Service Function Chaining

I. Background

Network related services are blooming in the recent years and have become part of people's daily life. After years of development, audio and video contents make up a large portion of the Internet portion. These data may be related to large streaming media (e.g., Netflix, Youtube) or social media (e.g., Facebook, Instagram).

Multicasting is a fundamental networking functionality that enables efficient data distribution and is used in many network applications, such as audio distribution, video conference, streaming TV and querying in large datacenters.

To ensure stable connection and secured multicasting in these scenarios, some network functions (e.g., proxies, firewalls) are needed. With the technology of Network Function Virtualization (**VNF**), these network functions can be implemented by software running on virtual machines (**VM**) in a Software Defined Networks (**SDN**). Network functions may be implemented in different software on different VMs to enhance flexibility, which leads to Service Function Chaining (**SFC**).

Traditional IPv4 and IPv6 multicasting protocols are limited by their decentralized routing policy. Routing policies in SDN, however, are centralized and kept inside a special controller (**SDN controller**) whose job is to manage the switching table of each switches in the SDN. With centralized intelligence, multicasting in SDN may outperform traditional multicasting protocols.

II. Purpose

NFV may highly increase the flexibility of an SDN, but it also adds complexity to multicast routing problem. In general cases, a multicast packet is required to be processed by several network functions in a specific order before reaching any of its destinations. These network functions may be implemented on different NFV-enabled servers. Therefore, both transmission and computing resource should be taken into consideration when constructing SFCs in order to prevent network congestion or server overload.

Bangbang Ren et al. studied the construction method of SFC in their paper "Embedding Service Function Tree With Minimum Cost for NFV-Enabled Multicast" in 2019, but they did not propose a good method of balancing computing resource usage and bandwidth usage. Zichuan Xu et al. proposed an algorithm to find an approximate minimum cost multicast tree in their paper “Efficient NFV-Enabled Multicasting in SDNs" published in 2019, but they restricted SFC to be implemented on only 1 server. Moreover, their algorithm has a relatively large time complexity[1] and might be impractical for real applications. Haozhe Ren et al. discussed the multicasting problem with SFC constraints in "Efficient Algorithms for Delay-Aware NFV-Enabled Multicasting in Mobile Edge Clouds With Resource Sharing" in 2020. However, their optimization is about operation cost, and the time complexity is quite large[2].

In this research, we studied NFV-enabled multicasting problems in SDN supporting SFC. the major goal is to develop a multicast routing algorithm that maximizes multicast requests by avoiding network congestion, subject to network resource constraints including NFV server computational capability and link bandwidth. We developed two algorithms: **Find_SFC** and **Chen’s_MT** to construct multicast trees. They have lower time complexity and better approximate ratios than previous studies.

III. Method

In this research, we first constructed the system mode of NFV-enabled SDN and define the problems rigorously. Later, we transformed the problem into minimum cost problem by designing a cost model.

After the cost model is designed, we developed two algorithms: Find_SFC and Chen’s_MT. The first is used for finding the most efficient SFC candidates, and the second is used for finding an approximate minimum cost multicast tree. These algorithms are proved effective with low time complexity.

IV. Result

In this research, we constructed the system model of NFV-enabled SDN. Let $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ is an SDN, where $\boldsymbol{V}$ is a set of SDN-enabled switches, and $\boldsymbol{E}$ is a set of links between these switches. Some switches are connected with NFV-enabled servers, denoted as $\boldsymbol{V_s}$, and $\boldsymbol{V_s} \subset \boldsymbol{V}$. Each link e in $\boldsymbol{E}$ has its link bandwidth, B_e. Similarly, each server v_s in $\boldsymbol{V_s}$ has its own computing capacity, denoted as C_v.

---

[1] $O(|V|^3|V_s|^M)$, where $|V|$ is the number of switches, $|V_s|$ is the number of NFV-enabled servers and M is the number of servers put in work.

[2] $\approx O((|V|^3 \log|V|) + (L_k|V|)^i\ |D_k|^{2i})$, where $|V|$ is the number of switches, L_k is the number of VNF required, i is the level of directed Steiner Tree and $|D_k|$ is the number of destinations.

A multicast request ***r*** is sent to the SDN controller whenever a switch requests for multicasting. The multicast request consists of four components: (1) multicast source s (2) multicast destinations D (3) demanded bandwidth b (4) service function chain SC and is denoted as $\boldsymbol{r} = (s, \boldsymbol{D}, b, \boldsymbol{SC})$, and the k^{th} multicast request a SDN controller received may be denoted as $\boldsymbol{r}_{\mathrm{k}} = (s_k , \boldsymbol{D}_k , b_k , \boldsymbol{SC}_k)$. The service function is a sequence of network functions that are implemented by NFV-enabled servers in an SDN.

Later, we define the problem rigorously as follows:

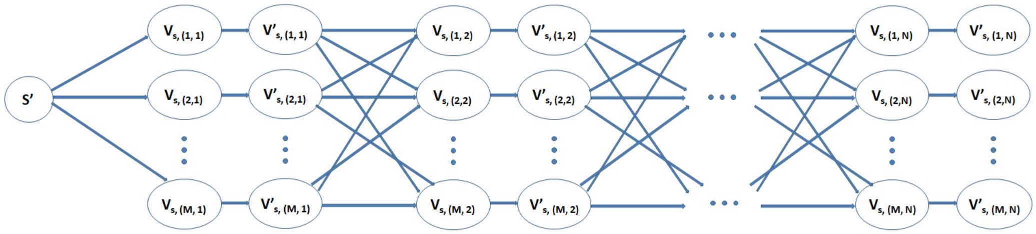
> Problem 1: Assume the transmission links and the NFV-enabled servers in an SDN ***G*** have limited capacity. Multicast requests for creating a multicast group or closing a multicast group arrive over time. This problem aims to admit as many multicast requests without knowing future multicast requests while meeting current computing and transmission constraints. Multicast destinations are not allowed to join or leave the multicast group after the multicast tree is constructed.
>
> Problem 2: In reality, multicast group members do join or leave over time. If multicast trees have to be reconstructed every time when “join” or “leave” happens, it is a waste of SDN controller’s computation resource. This problem aims to admit as many multicast requests while meeting current resource constraints, and is able to handle group member joining or leaving.

To solve problem 1, the algorithm has to distribute the limited resource (both server computing power and transmission bandwidth) in a reasonable way. We assign each unit of computing power and transmission bandwidth with some virtual cost. This turns the problem into finding the multicast tree with minimum cost. Therefore, we assign the unit cost of computing power $c_v(k)$ and $c_e(k)$ as:

$$\mathrm{c_v(k)} = (\mathrm{e}^{\alpha\left(\frac{\mathrm{C_v - AC_v(k)}}{\mathrm{C_v}}\right)} - 1) \qquad\qquad \mathrm{c_e(k)} = (\mathrm{e}^{\beta\left(\frac{\mathrm{B_e - AB_e(k)}}{\mathrm{B_e}}\right)} - 1)$$

where $AC_v(k)$ is the amount of available computing capability of server v and $AB_e(k)$ be the amount of available bandwidth of link e when multicast request r_k arrives and α, β are constant system parameters.

Later, we construct an auxiliary graph $\mathbf{G_{SFC}}$ with algorithm **Find_SFC** described in the research paper, an example of $\mathbf{G_{SFC}}$ is shown as follows:



We then find M shortest paths $\boldsymbol{P_i}$, each connecting a $v'_{s,(i,n)}$, i = 1, 2, … M and s' in an efficient method. The paths $\boldsymbol{P_i}$ represent the most efficient (minimum cost) ways of implementing SFC in the SDN network.

After the SFC candidates $\boldsymbol{P_i}$ are found, induce algorithm **Chen's_MT** described in the research paper to find an approximate minimum cost tree $\boldsymbol{MT_k}$. With detailed proofs, $\boldsymbol{MT_k}$ is indeed a multicast tree for multicast request $\boldsymbol{r_k}$ with SFC embedded. $\boldsymbol{MT_k}$ has a good approximate ratio $2\cdot(1 - 1 / |\mathbf{D_k}|)$ to the optimal multicast tree, where $|\mathbf{D_k}|$ is the number of destinations of $\mathbf{r_k}$. The time complexity of constructing $\boldsymbol{MT_k}$ is $\mathbf{O}(|\mathbf{V_s}|^2|\mathbf{SC_k}| + |\mathbf{V_s}||\mathbf{E}|\log_2(|\mathbf{V}|) + |\mathbf{D_k}||\mathbf{V_s}|^2) \approx \mathbf{O}(|\mathbf{D_k}||\mathbf{V_s}|^2)$, where $|\mathbf{V_s}|$ is the number of total NFV-enabled server and $|\mathbf{SC_k}|$ is the number of network function required. This is extremely smaller than previously existed studies.

V. Conclusion

In this research, we studied the algorithms of NFV-enabled multicast routing supporting SFC in SDN with limited transmission and computation resource. We developed cost models to transfer the original problem into finding minimum cost multicast trees. Algorithm **Find_SFC** is developed to efficiently discover SFC candidates, and algorithm **Chen's_MT** finds the approximate minimum multicast tree with a good approximate ratio. The total time complexity outperforms existed algorithms, while the approximate ratio is comparatively good.