Service Function Chain Composition and Mapping in NFV-enabled Networks

Meng Wang, Bo Cheng, Biyi Li, Junliang Chen
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
Beijing, P.R. China
{*mengwang, chengbo, lbyi0402, chjl*}*@bupt.edu.cn*

***Abstract*—Network Function Virtualization (NFV) is a new network paradigm that decouples network functions from dedicated hardware. Network services in NFV are deployed as service chains, also known as Service Function Chains (SFCs). SFC consists of an ordered set of Virtual Network Functions (VNFs). One of the main challenge when deploying SFC is to efficiently make use of the resource. In this paper, we focus on the SFC composition and mapping considering resource optimization. We formulate the SFC composition and mapping problem as a weighted graph matching problem. Then we propose a Hungarian based algorithm to solve the SFC composition and mapping problem in a coordinated way.**

***Keywords*-Network Function Virtualization; Service Function Chain; Composition; Mapping; Graph Matching;**

I. Introduction

With the development of network services, enterprise network nowadays is composed of multiple network functions and how to deploy them in an efficient way becomes a key challenge. Operators are bound by specific hardware since the tightly coupled deployment of network functions on the physical resource. Network Function Virtualization (NFV) [1] is a novel network paradigm that decouples network functions from the hardware. By allowing various network functions to be virtualized, the Capital Expenditure (CAPEX) and Operating Expense (OPEX) can be decreased significantly [2]. Based on the requirements and dependencies of network applications, network services are described as service chains. The service chain, also known as Service Function Chain (SFC), consists of an ordered set of Virtual Network Functions (VNFs).

For SFC, it runs on physical nodes and consists of several VNFs. One of the main challenges when deploying SFC is resource optimization. This challenge is also known as NFV resource allocation problem [3] consisting of three subproblems: VNF chain composition, VNF forwarding graph embedding and VNF scheduling. In this paper, the SFC composition and mapping correspond to the VNF chain composition and VNF forwarding graph embedding, respectively. In detail, there are multiple requirements and dependencies in a SFC Request (SFCR). Therefore, we need to efficiently compose the SFCR as a request graph, also known as the VNF Forwarding Graph (VNF-FG) [4]. After the SFC composition, we get a suitable VNF-FG. Then, we need to map this VNF-FG to the physical network. As VNF-FG and physical network are dynamic, different mapping methods can result in different resource consumption.

The requirements and dependencies of network applications complicate the deployment of SFC. In this paper, we focus on the SFC composition and mapping problem, which has received increasing attention from both academic and industry. There are different kinds of SFC composition and mapping algorithms in the existing works [5], [6], [7]. However, there are still some problems remaining to be solved. Firstly, most of the existing works propose heuristic algorithms to solve the SFC mapping problem. However, heuristic approaches iteratively solve the problem and it can affect the quality of the solutions and increase the time to find a suboptimal solution. Secondly, more existing works solve the SFC composition and mapping problem in a separate way, resulting in low efficiency.

Given these facts, we optimize the SFC composition and mapping problem. Firstly, we formulate this problem as a Weighted Graph Matching Problem (WGMP). In detail, we describe SFCR and physical network as VNF-FG and Physical Network Graph (PNG) and compute the similarity matrix between them. Then we compute the dependency matrix of VNFs. Based on the similarity matrix and dependency matrix, we solve the SFC composition. And we propose a Hungarian based algorithm to map SFC. In summary, the main contributions are as follows:

- Formulate the SFC composition and mapping problem as a WGMP. We use an eigendecomposition based approach to compute the similarity matrix between VNF-FG and PNG. Our proposed solutions can run in polynomial time and optimize resource consumption.
- Propose a Hungarian based algorithm to solve the SFC composition and mapping in a coordinated way. We compute the dependencies between VNFs according to the requirements and priorities in SFC. Based on the similarity matrix and dependency matrix, we solve the SFC composition and mapping problem.

The rest of this paper is organized as follows. Section II discusses the related works. Section III describes and formulates the problem. Section IV describes our preliminary idea. Finally, Section V concludes this study.

II. Related Works

For resource allocation in NFV, Herrera *et. al.* [3] divide the resource allocation problem into three subproblems: VNF chain composition, VNF forwarding graph embedding and VNF scheduling. The VNF chain composition and VNF forwarding graph embedding correspond to the SFC composition and mapping we are concerned with.

In [5], [7], authors solve the SFC composition and mapping problem in a coordinated way. However, most of these approaches use heuristic algorithms to iteratively solve the problem and increase the time to find a suboptimal solution.

Jemaa *et. al.* [6] propose an eigen based approach that can efficiently solve the SFC mapping problem. However, they do not take into account the SFC composition but instead takes a predetermined VNF-FG as the input.

In summary, most of the existing works use heuristic algorithms to solve the SFC composition and mapping problem. In our previous work, we focus on the SFC composition and mapping problem considering availability guarantee and resource optimization.

In this paper, we optimize the SFC composition and mapping problem in a novel way. We formulate this problem as a WGMP and propose a Hungarian based algorithm to optimize resource consumption.

III. Problem Statement

In this section, we describe the SFC composition and mapping problem. Then, we formulate it as a WGMP.

A. Problem Description

1) SFC Composition: As Fig. 1(a) shows, R indicates the relative flow ratio of two VNFs (e.g. R = 40% between VNF_1 and VNF_5). VNF_1 (load balancer) divides data into two streams. 60% of incoming traffic is forwarded to VNF_4, and 40% to VNF_5. The blue dotted line indicates network flow. And the red dotted line indicates dependencies of VNFs. For example, the red dotted line from VNF_3 to VNF_2, indicating that VNF_2 must be executed before VNF_3.

As Fig. 1(b) shows, since there are no obvious dependencies between some VNFs, SFCR can generate two different VNF-FGs. For example, there is no qualitative requirement for the order of VNF_2 and VNF_5.

Next, we define the relative data rate of VNF in a VNF-FG. We assume that VNF_1 executes video encoding function. And VNF_1 requires the processing capabilities of a 500 MHz CPU to encode 100 MBit/s. Therefore, the relative data rate of VNF_1 is 500 * 100 = 50/Gbps and the total data rate is D = 50/Gbps * 1Gbps = 50.

2) SFC Mapping: By calculating the total data rate of each VNF, we can conclude that different VNF-FGs have different total data rates. As Fig. 1(b) shows, the data rates of VNF_2 and VNF_5 are different in the red dotted box. Therefore, the optimal way is to map the VNF-FG with low data rate to the physical network, as shown in Fig. 1(c).

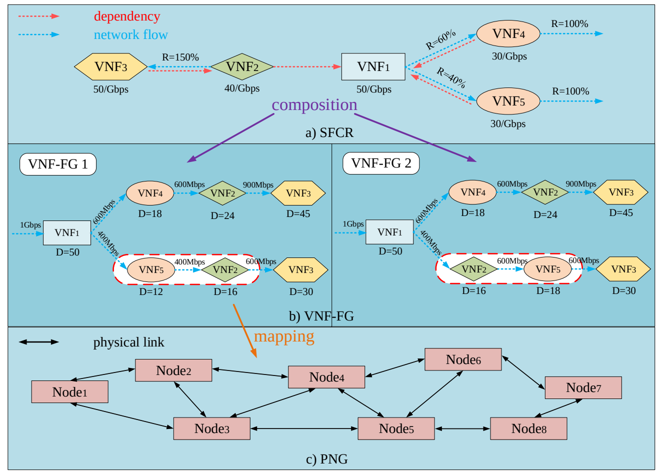

Figure 1. SFC Composition and Mapping in NFV-enabled Networks.

However, different mapping methods can lead to different resource consumption. Therefore, we need to find a VNF-FG with the lowest resource consumption and map it to the physical network with an efficient mapping algorithm.

B. Problem Formulation

In this subsection, we formulate the SFC composition and mapping problem in NFV-enabled networks. The used notations can be found in Table I.

Table I
Basic Notations in This Paper.

Symbol	Definition
VNF-FG	
$V = \{v_1, v_2 \cdots\}$	the set of VNFs.
$E = \{e_{pq} \cdots\}$	the set of logic links.
$e_{pq} = (v_p, v_q)$	the logic link between v_p and v_q.
PNG	
$N = \{n_1, n_2 \cdots\}$	the set of physical nodes.
$L = \{l_{ij} \cdots\}$	the set of physical links.
$l_{ij} = (n_i, n_j)$	the physical link between n_i and n_j.
Resource	
$cpu_{v_p}, mem_{v_p}, bw_{e_{pq}}$	the CPU, memory and bandwidth cost.
$C^{cpu}_{n_i}, C^{mem}_{n_i}, C^{bw}_{l_{ij}}$	the CPU, memory and bandwidth capacity.
Variables	
$\alpha_{v_p \subset n_i}$	whether v_p is mapped in n_i.
$\beta_{e_{pq} \subset l_{ij}}$	whether e_{pq} is mapped in l_{ij}.

1) VNF-FG: We define $F = (V, E)$ as a VNF-FG in SFC Request (SFCR). In this VNF-FG, V indicates the set of VNFs and E indicates the set of logical links between two VNFs. cpu_{v_p} and mem_{v_p} indicate the CPU and memory consumption of VNF $v_p \in V$, respectively. In addition, we define $bw_{e_{pq}}$ as the bandwidth consumption of logical link $e_{pq} \in E$.

2) Physical Network Graph (PNG): The physical network can be formulated as a bipartite graph $P = (N, L)$. In this paper, we name this graph as Physical Network Graph (PNG). N and L are the set of physical nodes and physical links, respectively. In addition, $C^{cpu}_{n_i}$ and $C^{mem}_{n_i}$ indicate the CPU and memory capacity of each physical node $n_i \in N$. And we use $C^{bw}_{l_{ij}}$ to indicate the bandwidth capacity of physical link $l_{ij} \in L$.

3) Objectives: In this subsection, we use binary variable $\alpha_{v_p \subset n_i}$ to indicate the mapping status:

$$\alpha_{v_p \subset n_i} = \begin{cases} 1 & \text{if } v_p \text{ is mapped in } n_i \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

We take CPU, memory and bandwidth constraints into consideration. In detail, the CPU consumption of all VNFs placed in the same physical node cannot exceed the CPU capacity of this physical node. Therefore, the CPU constraints can be formulated as:

$$\sum_{p=1}^{|V|} cpu_{v_p} \cdot \alpha_{v_p \subset n_i} \leq C^{cpu}_{n_i} \quad (2)$$

Similarly, the memory constraints are:

$$\sum_{p=1}^{|V|} mem_{v_p} \cdot \alpha_{v_p \subset n_i} \leq C^{mem}_{n_i} \quad (3)$$

We use $\beta_{e_{pq} \subset l_{ij}}$ to describe whether logical link e_{pq} is mapped in physical link l_{ij}:

$$\beta_{e_{pq} \subset l_{ij}} = \begin{cases} 1 & \text{if } e_{pq} \text{ is mapped in } l_{ij} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Therefore, the bandwidth constraints are:

$$\sum_{p=1}^{|V|} bw_{e_{pq}} \cdot \beta_{e_{pq} \subset l_{ij}} \leq C^{bw}_{l_{ij}} \quad (5)$$

Our goal is to optimize the resource consumption:

$$\min \sum_{p=1}^{|V|} (\alpha_{v_p \subset n_i} + \beta_{e_{pq} \subset l_{ij}}) \quad (6)$$

s.t. Eq. 1 to Eq. 5.

Different from the traditional approaches, we formulate SFC composition and mapping problem as a WGMP to reduce resource consumption such as the used physical nodes and physical links.

IV. Preliminary Ideas

In this section, we propose a preliminary idea based on graph matching. While Weighted Graph Matching Problem (WGMP) [8] is an old idea, it has not been widely used in the SFC composition and mapping in NFV-enabled networks.

Firstly, we compute the adjacent matrices of VNF-FG and PNG. Secondly, we use the eigendecomposition based approach to compute the similarity between the two graphs. Thirdly, we get the similarity matrix and compute the dependency matrix according to the priorities between VNFs. Finally, we solve the SFC composition based on the similarity matrix and dependency matrix. After generating an optimal VNF-FG, we map it based on Hungarian method.

Algorithm 1: Preliminary Ideas

Input: The VNF-FG: $F = (V, E)$;
The PNG: $P = (N, L)$;

Output: The mapping result: M_{res};

Step1: Compute adjacent matrices A_F and A_P:

$$A_F = \begin{cases} f_{pp} = cpu_{v_p} \\ f_{pq} = bw_{e_{pq}} \end{cases} \quad A_P = \begin{cases} p_{ii} = C^{cpu}_{n_i} \\ p_{ij} = C^{bw}_{l_{ij}} \end{cases}$$

Step2: Minimum the distance between VNF-FG and PNG. Then, compute the eigen vectors.

Step3: Compute the similarity matrix and the dependency matrix.

Step4: SFC composition based on the similarity matrix and the dependency matrix. SFC mapping based on Hungarian method.

---

1) Adjacent Matrix (Step 1): At Step 1, A_F and A_P are used to indicate the adjacent matrices of VNF-FG and PNG. f_{pp} indicates the CPU consumption of VNF v_p. f_{pq} indicates the bandwidth consumption between VNF v_p and v_q (so is p_{ii} and p_{ij}). We use Dijkstra method to compute the weight between nodes (or VNFs) that are not directly connected.

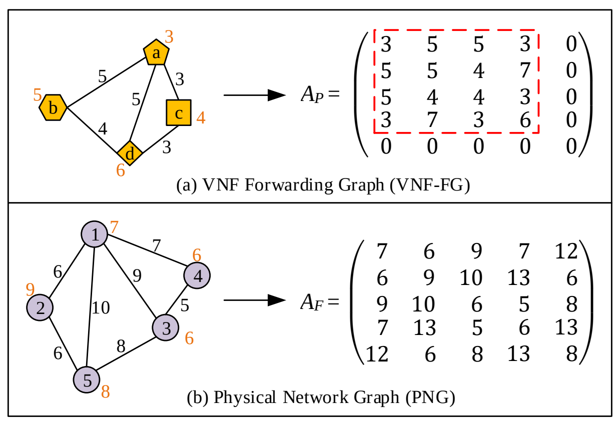

Figure 2. (a) VNF-FG; (b) PNG.

For example, as Fig. 2(a) shows, the adjacency matrix $(4 * 4)$ of VNF-FG is in the red dotted line box. Since the WGMP requires two weighted graphs with the same size, we extend VNF-FG to a $5 * 5$ matrix (the newly added elements are all zeros). In addition, PNG is a $5 * 5$ matrix, as shown in Fig. 2(b).

2) Distance and Eigen Vectors (Step 2): The goal of WGMP is to minimize the distance of VNF-FG and PNG.

Based on [8], the distance can be defined as:

$$J(\Phi) = \sum_{i=1}^{n} \sum_{j=1}^{n} (p(n_i, n_j) - f(\Phi(n_i), \Phi(n_j)))^2 \quad (7)$$

Where $p(n_i, n_j)$ indicates the weight between node n_i and n_j. Φ indicates the one-to-one correspondence between VNF and node. In detail, we assume that $(\Phi(n_i), \Phi(n_j)) = (v_p, v_q)$. Then, we can use $f(\Phi(n_i), \Phi(n_j)) = f(v_p, v_q)$ to indicate the weight between VNF v_p and v_q.

The goal of WGMP is to minimum the distance $J(\Phi)$. Therefore, the matching problem can be reformulated as:

$$\min_M \left\| A_P - M A_F M^{\mathrm{T}} \right\| \quad (8)$$

Where permutation matrix M indicates the mapping function Φ. A_F and A_P are the adjacency matrices of two weighted graphs.

According to [8], WGMP can be reformulated as:

$$tr(M^{\mathrm{T}} \bar{U}_P \bar{U}_F^{\mathrm{T}}) \leq n \quad (9)$$

Where $\bar{U}_F$ and $\bar{U}_P$ indicate the eigenvector matrices of A_F and A_P, respectively.

3) Similarity Matrix and Dependency Matrix (Step 3): According to Eq. 9, we can get the permutation matrix M:

$$M = \bar{U}_P \bar{U}_F^{\mathrm{T}} \quad (10)$$

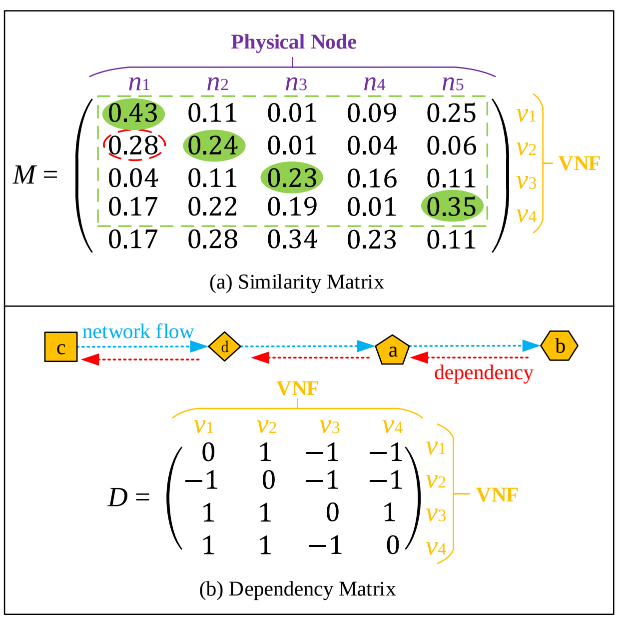

Figure 3. (a) Similarity Matrix M; (b) Dependency Matrix D.

As Fig. 3(a) shows, data in green dotted line box represents the similarity between physical nodes and VNFs. For example, m_{11} represents the similarity between n_1 and v_1.

As Fig. 3(b) shows, we compute a $4{*}4$ dependency matrix based on the priorities between VNFs. We use $d_{ij} = 1$ indicates that the execution order of VNFs is allowed (otherwise, it is not allowed). For example, $d_{12} = 1$ indicates that VNF_a executed before VNF_b is allowed.

4) SFC Composition and Mapping (Step 4): Based on the similarity matrix and dependency matrix, we can compose the SFC and get an optimal VNF-FG. Then we map this VNF-FG to the physical network.

As Fig. 3(a) shows, the optimal match should be that v_1 and v_2 are mapped to n_1, v_3 is mapped to n_3 and v_4 is mapped to n_5. However, the resource capacity of n_1 cannot satisfy v_1 and v_2 ($C_{n_1}^{cpu}$ = 7 while cpu_{v_1} = 3 and cpu_{v_2} = 5). Therefore, v_2 should be mapped to n_2 since the similarity between n_2 and v_2 is the next highest.

V. Conclusion

In this paper, we focus on SFC composition and mapping considering resource optimization. We formulate the problem as a WGMP and propose a Hungarian based algorithm. In future work, we further study resource optimization.

Acknowledgment

This work was supported in part by the National Key Research and Development Program of China No.2018YFB1003804, in part by the National Natural Science Foundation of China No.61772479, and in part by the BUPT Excellent Ph.D. Students Foundation under Grant CX2019214.

References

[1] Network functions virtualisation (nfv). [Online]. Available: https://portal.etsi.org/Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf

[2] 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 and Tutorials*, vol. 18, pp. 236–262, 2016.

[3] J. Gil-Herrera and J. F. Botero, "Resource allocation in nfv: A comprehensive survey," *IEEE Transactions on Network and Service Management*, vol. 13, pp. 518–532, 2016.

[4] E. I. NFV, "Network functions virtualisation (nfv): Terminology for main concepts in nfv, https://www.etsi.org/deliver/etsi_gs/nfv/001_099/003/01.02.01_60/gs_nfv003v010201p.pdf," 2014, eTSI, NFVGS.

[5] M. T. Beck and J. F. Botero, "Scalable and coordinated allocation of service function chains," *Computer Communications*, vol. 102, pp. 78–88, 2017.

[6] M. Mechtri, C. Ghribi, and D. Zeghlache, "A scalable algorithm for the placement of service function chains," *IEEE Transactions on Network and Service Management*, vol. 13, pp. 533–546, 2016.

[7] M. T. Beck and J. F. Botero, "Coordinated allocation of service function chains," *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2014.

[8] S. Umeyama, "An eigendecomposition approach to weighted graph matching problems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, pp. 695–703, 1988.