Multiple Service Chain Placement and Routing in a Network-enabled Cloud

Abhishek Gupta*, Brigitte Jaumard†, Massimo Tornatore*‡, and Biswanath Mukherjee*

*University of California, Davis, USA   †Concordia University, Canada   ‡Politecnico di Milano, Italy

Email: *{abgupta,mtornatore,bmukherjee}@ucdavis.edu †bjaumard@cse.concordia.ca ‡massimo.tornatore@polimi.it

Abstract—**Network Function Virtualization (NFV) aims to abstract the functionality of traditional proprietary hardware into software as Virtual Network Functions (VNFs), which can run on commercial off-the-shelf (COTS) servers. Besides reducing dependency on proprietary support, NFV helps network operators to deploy multiple services in a agile fashion. Service deployment involves placement and in-sequence routing through VNFs comprising a Service Chain (SC). Our study is the first to focus on the computationally-complex problem of multiple VNF SC placement and routing while considering VNF service chaining explicitly. We propose a novel column-generation model for placing multiple VNF SCs and routing, which reduces the computational complexity of the problem significantly. Our aim here is to determine the ideal NFV Infrastructure (NFVI) for minimizing network-resource consumption. Our results indicate that a Network-enabled Cloud (NeC) results in lower network-resource consumption than a centralized NFVI (e.g., Data Center) while avoiding infeasibility with a distributed NFVI.**

I. Introduction

Today's communication networks provide services through proprietary hardware appliances (e.g., network functions such as firewalls, NAT, etc.) which are statically configured to provide a service. Static configurations of network functions are difficult to deploy, modify, and upgrade. Further, with rapid evolution of applications in today's internet, networks require more agile and scalable service deployment. A rapid and flexible deployment of services, however, is not feasible with traditional configured services since they are physically embedded in the network.

Network Function Virtualization (NFV) [1] provides network operators with a solution. NFV propagates the idea of hardware functionality abstracted into software modules called Virtual Network Functions (VNFs). VNFs can be run on commercial-off-the-shelf (COTS) hardware such as servers and switches. VNF placement on COTS hardware avoids embedding functions in the network and makes service deployment agile and scalable. But VNF placement has resource requirements (e.g., CPU cores, RAM, storage) that need to be satisfied by hardware, same as Virtual Machines (VMs). Since VNFs essentially are VMs, they can be deployed in the cloud.

Cloud usually refers to massive compute and storage facilities offered by Data Centers (DCs). Here, we extend cloud to the network with a Network-enabled Cloud (NeC) [2] where network elements including elements from the underlying optical backbone network that are equipped with computing and storage resources can be part of the NFV Infrastructure
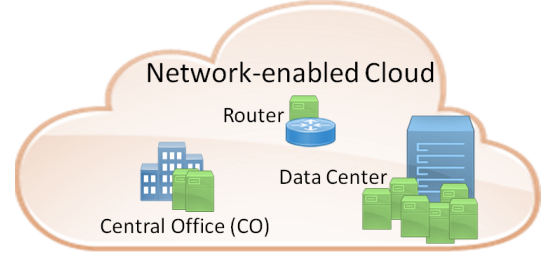

Fig. 1: A Network-enabled Cloud (NeC).

(NFVI) and are called "NFV Point of Presence" (NFV-PoP) such as router and Central Office (CO) in Fig. 1. A NeC will lead to better network resource utilization as traffic flows will take shorter paths by removing frequent redirection to DC for service. Any implementation of NFVI including NeC will rely heavily on the optical backbone network.

When network functions are configured to provide a service, we have a "Service Chain". The term "service chain" is used "to describe the deployment of such functions, and the network operator's process of specifying an ordered list of service functions that should be applied to a deterministic set of traffic flows" [3]. So, a "Service Chain" (SC) specifies a set of network functions configured in a specific order. With NFV, we have VNF SCs where VNFs are configured in a specific sequence to form a SC as shown in Fig. 2.

In this work, we solve the problem of multiple VNF SC placement and traffic-flow routing. This problem is difficult to scale and computationally complex which is a significant challenge for network operators as they aim to deploy NFV to provide more agile services while utilizing network resources efficiently. We propose a novel sophisticated column-generation model to solve this problem and find the ideal NFVI scheme. The relatively-small computation time of the model helps us to scale to large problem instances which is not possible with standard mathematical modeling.

The rest of the study is organized as follows. Section II gives an overview of major works to solve the VNF placement and routing problem. Section III formally describes the problem and its input parameters. We then describe our column-generation model in Section IV to solve the multiple VNF SC placement and routing problem. Section V uses illustrative examples to demonstrate the idea of an NeC. Section VI
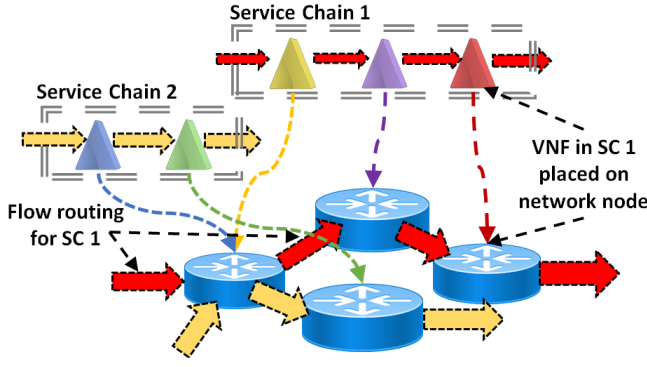
Fig. 2: Multiple VNF SC placement and routing.

concludes the study.

II. RELATED WORK

A number of studies exist on the VNF placement and routing problem. The authors of Ref. [4] study a hybrid deployment scenario with hardware middleboxes using an Integer Linear Program (ILP), but do not enforce VNF service chaining explicitly. Ref. [5] uses an ILP to study trade-offs between legacy and NFV-based traffic engineering but does not have explicit VNF service chaining. Ref. [6] models the problem in a DC setting using an ILP to reduce the end-to-end delays and minimize resource over-provisioning while providing a heuristic to do the same. Here too the VNF service chaining is not explicitly enforced by the model. Ref. [7] models the batch deployment of multiple chains using an ILP and develops heuristics to solve larger instances of the problem. However, they enforce that VNF instances of a function need to be on a single machine and restrict all chains to three VNFs. Our model does not impose such constraints, and we allow any VNF type to be placed on any NFV-PoP and any number of VNFs in a SC while service chaining VNFs for a SC explicitly.

Our previous work [8] solved the problem for a single VNF SC in a NeC. Scaling the model to multiple VNF SCs resulted in exponential time complexity. We now address this challenge by using a novel decomposition model (column generation) for multiple VNF SC placement and routing. Our objective is to minimize network-resource consumption in any NFV infrastructure (NFVI) where VNF SCs are deployed. And here, we make the case for a Network-enable Cloud (NeC) to be ideal for NFVI. Column generation allows us to place multiple VNF SCs in a relatively small amount of time; and to the best of our knowledge, it is the first to solve the multiple VNF SC placement and routing problem.

III. PROBLEM DESCRIPTION

An operator's network provides multiple services. Traffic demand for a service is satisfied by sequential traversal of VNFs comprising a SC for the service. Configuring VNFs to form an SC is an important problem but is not the focus of this work. Here, we assume that, given a service, the operator

has a pre-configured SC, i.e., the ordered sequence of VNFs is already known. With the knowledge of SC configurations for each service, the network operator has to provide for the service requirements of all traffic flows with minimal network resources. The problem lends itself to be modeled as an optimization problem where the objective is to reduce network-resource consumption by optimally placing and routing traffic through multiple VNF SCs. Fig. 2 depicts the multiple VNF SC placement and routing problem.

A. Problem Statement

Given a network topology, capacity of links, a set of network nodes with NFV support (NFV-PoP), traffic flows for source-destination (v_s, v_d) pairs, bandwidth demand of traffic flows for (v_s, v_d) pairs, set of VNFs, and the set of SCs, we determine the placement of VNFs and corresponding routing to minimize network-resource (bandwidth) consumption.

B. Input Parameters

VNFs are distinct from VMs in that they are more bandwidth intensive. This happens as VNFs are virtual instances of network functions, which operate at line rate. Also, since a number of physical network functions are compute intensive, we assume VNF behaviour to be compute intensive. Each VNF has CPU core requirement to process unit traffic throughput. We assume that CPU core requirements of VNFs increase linearly with traffic throughput.

- $G = (V, L)$: Physical topology of the optical backbone network; V is set of nodes and L is set of links.
- $V^{\text{NFV}} \subseteq V$: Set of NFV-PoP nodes.
- F, indexed by f: Set of VNFs.
- n^{CORE}: Number of CPU cores present per NFV-PoP node.
- n_f^{CORE}: Number of CPU cores for function f to give unit throughput.
- C: Set of chains, indexed by c.
- n_c: Number of VNFs in SC c.
- $\mathcal{SD}$: Set of source-destination (v_s, v_d) pairs.
- D_{sd}^c: Traffic demand between v_s and v_d for SC c.
- $\sigma_i(c)$: ID of ith VNF in SC c where $f_{\sigma_i(c)} \in F$.

As discussed in Section II, our previous model [8] did not scale for multiple SCs and could not enforce the number of SC instances to be deployed. Hence, we have utilized a Decomposition Model ('Column Generation'), which scales for multiple VNF SC deployment and traffic routing.

IV. DECOMPOSITION MODEL

Each SC, denoted by c, is characterized by an ordered set of n_c functions:

$$[\text{SC } c] \qquad f_{\sigma_1(c)} \prec f_{\sigma_2(c)} \prec \cdots \prec f_{\sigma_{n_c}(c)}. \qquad (1)$$

Each deployment of SC c is defined by a set of VNF locations, and a set of paths, from the location of first VNF to location of last VNF.

Our decomposition model relies on a set of *chain configurations* where each configuration is associated with a potential provisioning of a SC c and a potential node placement of its

functions. Let Γ be the set of configurations, and Γ_c be the subset of configurations associated with service chain $c \in C$:

$$\Gamma = \bigcup_{c \in C} \Gamma_c.$$

Due to its definition, the number of configurations is exponential. Given this consideration, we find the problem to fit naturally in the column-generation framework [9].

Column generation (**CG**) is a decomposition technique, where the problem (called Master Problem - **MP**) to be solved is divided into two sub-problems: restricted master problem (**RMP**) (selection of the best configurations) and pricing problems (**PP_SC**$(c))_{c \in C}$ (configuration generators for each chain). The CG process involves solving the **RMP**, querying the dual values of **RMP** constraints, and using them for **PP_SC**(c) objective. Each improving solution (i.e., with a negative reduced cost) of **PP_SC**(c) is added to **RMP**, and previous step is repeated until optimality condition is reached [9], with the **PP_SC**(c) explored in a round robin fashion.

A chain configuration is characterized by the following parameters:

- Location of the functions: $a_{vf}^{\gamma} = 1$ if $f \in c$ is located in v in configuration; 0 otherwise.
- Connectivity of the locations: path from the location of current VNF to next VNF in SC c. If link ℓ is used in the path from the location of $f_{\sigma_i(c)}$ to the location of $f_{\sigma_{i+1}(c)}$, then $b_{i\ell}^{\gamma} = 1$; 0 otherwise.

A. Restricted Master Problem (**RMP**)

RMP selects the best $\gamma \in \Gamma_c$ for each SC c. Also it finds a route from v_s (source) to first VNF of c and from last VNF of c to v_d (destination).
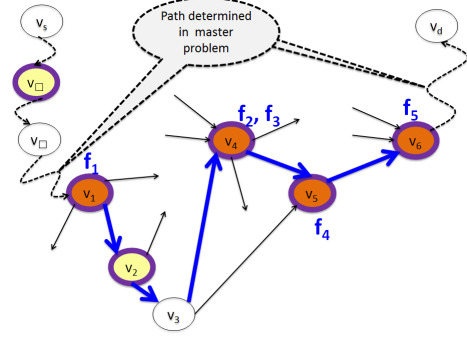
An illustration of the constraint splitting between **RMP** and **PP_SC**(c) is depicted in Fig. 3. Nodes circled in purple are NFV-PoP, yellow nodes do not host VNFs at present but have NFV support, and orange nodes currently host VNFs. Figure 3(a) has f_1 located at v_1. When a different configuration is selected in Fig. 3(b) and f_1 is located at v_2, then **RMP** finds the path from v_s to location of f_1. Similarly, **RMP** finds the path from last VNF to v_d, i.e., f_5 to v_d here.
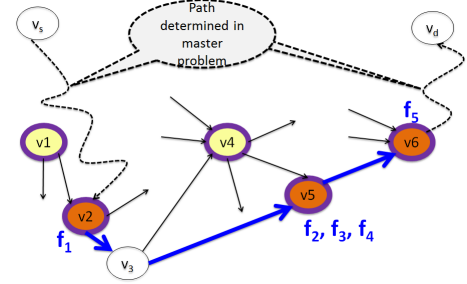
Variables:

- $z_{\gamma} = 1$ if configuration γ is selected; 0 otherwise.
- $x_v^f = 1$ if f is located in v; 0 otherwise.
- $y_{\ell}^{\text{first}(c),sd} = 1$ if ℓ is on path from v_s to location of first VNF in c; 0 otherwise.
- $y_{\ell}^{\text{last}(c),sd} = 1$ if ℓ is on path from location of last VNF in c to v_d; 0 otherwise.

Objective: Minimize bandwidth consumed:

$$\min \sum_{c \in C} \sum_{\gamma \in \Gamma_c} \underbrace{\sum_{\ell \in L} \overbrace{\left(\sum_{(s,d) \in \mathcal{SD}} D_{sd}^c \right)}^{\text{Overall traffic using } c} \sum_{i \in I} b_{i\ell}^{\gamma} z_{\gamma}}_{\text{COST}_{\gamma}} +$$

(a) A first configuration (γ_1) for c

(b) A second configuration (γ_2) for c

Fig. 3: Two configuration examples for chain $c = (f_1 \prec f_2 \prec f_3 \prec f_4 \prec f_5)$.

$$\sum_{c \in C} \sum_{\ell \in L} \sum_{(s,d) \in \mathcal{SD}} D_{sd}^c \left(y_{\ell}^{\text{first}(c),sd} + y_{\ell}^{\text{last}(c),sd} \right). \quad (2)$$

Total bandwidth consumed in placing multiple SCs depends on configuration γ selected for each SC c. Each γ for c locates VNFs of c and gives the route to traverse these VNF locations. So, bandwidth consumed when going from v_s to v_d and traversing the SC depends on selected γ.

Constraints:

$$\sum_{\gamma \in \Gamma_c} z_{\gamma} = 1 \qquad\qquad c \in C \qquad (3)$$

$$\sum_{\gamma \in \Gamma} \sum_{v \in V^{\text{NFV}}} a_{vf}^{\gamma} z_{\gamma} \geq 1 \qquad\qquad f \in F \qquad (4)$$

$$\sum_{c \in C} \sum_{\gamma \in \Gamma_c} \sum_{f \in F} n_f^{\text{CORE}} \sum_{(v_s,v_d) \in \mathcal{SD}} D_{sd}^c a_{vf}^{\gamma} z_{\gamma} \leq n^{\text{CORE}}$$
$$v \in V^{\text{NFV}} \qquad (5)$$

$$\sum_{c \in C} \sum_{(v_s,v_d) \in \mathcal{SD}} D_{sd}^c \left(y_{\ell}^{\text{first}(c),sd} + y_{\ell}^{\text{last}(c),sd} + \sum_{\gamma \in \Gamma} \sum_{i=1}^{n_c-1} b_{i\ell}^{\gamma} z_{\gamma} \right)$$
$$\leq \text{CAP}_{\ell} \qquad\qquad \ell \in L \qquad (6)$$

$$\sum_{\gamma \in \Gamma} a_{vf}^{\gamma} z_{\gamma} \geq x_v^f \qquad\qquad f \in F, v \in V^{\text{NFV}} \qquad (7)$$

$$\sum_{\gamma \in \Gamma} a_{vf}^{\gamma} z_{\gamma} \leq M x_v^f \qquad\qquad f \in F, v \in V^{\text{NFV}}. \qquad (8)$$

Constraints (3) guarantee that we select exactly one γ for SC c and forces c to have a single instance. Each γ is associated

with a set of a_{vf}^γ (from **PP_SC**(c)) required to be consistent with x_v^f in **RMP**, which is resolved by Eqs. (7) and (8).

We also ensure that each VNF f to be deployed across SCs is placed on at least one node using Eq. (4). While placing f in a NFV-PoP node, we need to ensure that this node has sufficient CPU cores. This is done using Eq. (5) with consideration for increase in compute resource demand due to traffic increase.

Eq. (6) enforces link-capacity constraints for the complete route for SC c from v_s to v_d ($\forall (v_s, v_d) \in \mathcal{SD} : D_{s,d}^c > 0$).

Route from v_s to first function location:

$$\sum_{\ell \in \omega^+(v_s)} y_\ell^{\text{first}(c),sd} = 1 - x_{v_s}^{\text{first}(c)} \qquad c \in C,$$
$$(v_s, v_d) \in \mathcal{SD} : D_{sd}^c > 0 \qquad (9)$$

$$\sum_{\ell \in \omega^-(v)} y_\ell^{\text{first}(c),sd} \geq x_v^{\text{first}(c)} \qquad c \in C,$$
$$(v_s, v_d) \in \mathcal{SD} : D_{sd}^c > 0, v \in V^{\text{NFV}} \setminus \{v_s\} \quad (10)$$

$$\sum_{\ell \in \omega^+(v)} y_\ell^{\text{first}(c),sd} - \sum_{\ell \in \omega^-(v)} y_\ell^{\text{first}(c),sd} = -x_v^{\text{first}(c)}$$
$$c \in C, (v_s, v_d) \in \mathcal{SD} : D_{sd}^c > 0, v \in V^{\text{NFV}} \setminus \{v_s\} \quad (11)$$

$$\sum_{\ell \in \omega^+(v)} y_\ell^{\text{first}(c),sd} - \sum_{\ell \in \omega^-(v)} y_\ell^{\text{first}(c),sd} = 0$$
$$c \in C, (v_s, v_d) \in \mathcal{SD} : D_{sd}^c > 0,$$
$$v \in V \setminus (V^{\text{NFV}} \cup \{v_s\}). \qquad (12)$$

We assume that an unique route exists from v_s to first VNF location. This is imposed by selecting exactly one outgoing link from v_s unless first VNF is located at v_s. We account for these scenarios using Eq. (9). To find the route from v_s to first VNF, flow conservation needs to be enforced at the intermediate nodes which may or may not have NFV support. Eqs. (11) and (12) enforces flow-conservation constraints at nodes with and without NFV support, respectively.

Eq. (10) ensures an incoming link to node hosting first VNF, unless that node is v_s.

Route from last function location to v_d:

$$\sum_{\ell \in \omega^-(v_d)} y_\ell^{\text{last}(c),sd} = 1 - x_{v_d}^{\text{last}(c)} \qquad c \in C,$$
$$(v_s, v_d) \in \mathcal{SD} : D_{sd}^c > 0 \qquad (13)$$

$$\sum_{\ell \in \omega^+(v)} y_\ell^{\text{last}(c),sd} \geq x_v^{\text{last}(c)} \qquad c \in C,$$
$$(v_s, v_d) \in \mathcal{SD} : D_{sd}^c > 0, v \in V^{\text{NFV}} \setminus \{v_d\} \quad (14)$$

$$\sum_{\ell \in \omega^+(v)} y_\ell^{\text{last}(c),sd} - \sum_{\ell \in \omega^-(v)} y_\ell^{\text{last}(c),sd} = x_v^{\text{last}(c)}$$
$$c \in C, (v_s, v_d) \in \mathcal{SD} : D_{sd}^c > 0, v \in V^{\text{NFV}} \setminus \{v_d\} \quad (15)$$

$$\sum_{\ell \in \omega^+(v)} y_\ell^{\text{last}(c),sd} - \sum_{\ell \in \omega^-(v)} y_\ell^{\text{last}(c),sd} = 0$$
$$c \in C, (v_s, v_d) \in \mathcal{SD} : D_{sd}^c > 0,$$
$$v \in V \setminus (V^{\text{NFV}} \cup \{v_d\}). \qquad (16)$$

Eqs. (13), (15), (16), and (14) enforce same functionality as Eqs. (9), (11), (12), and (10), respectively, but on route from location of last VNF to v_d.

B. Pricing Problem

Each SC c ($c \in C$) is associated with a pricing problem. The number of pricing problems to be solved equals the number of SCs to be deployed.

Pricing problem **PP_SC**(c) generates: *(i)* A set of locations for VNFs of c; and *(ii)* a sequence of paths from the location of VNF $f_{\sigma_i(c)}$ to the location of VNF $f_{\sigma_{i+1}(c)}$, for $i = 1, 2, \ldots, n_c - 1$ for chain c. Each solution that generated by **PP_SC**(c) with a negative reduced cost, leads to a new potential γ for c of interest.

Let $u_c^{(3)} \lesseqgtr 0, u_f^{(4)} \geq 0, u_v^{(5)} \geq 0, u_{vf}^{(7)} \geq 0, u_{vf}^{(8)} \geq 0$ be the values of the dual variables associated with constraints (3), (4), (5), (7), and (8), respectively.

Variables:
- $a_{vf} = 1$ if one occurrence of VNF f is located in $v \in V^{\text{NFV}}$; 0 otherwise.
- $b_{i\ell}^\gamma = 1$ if ℓ is on the path from location of $f_{\sigma_i(c)}$ to location of $f_{\sigma_{i+1}(c)}$; 0 otherwise.

Objective: Minimize reduced cost of variable z_γ:

$$[\textbf{PP\_SC}(c)] \quad \text{RED\_COST}_\gamma = \text{COST}_\gamma - \sum_{f \in F} u_f^{(4)} \sum_{v \in V} a_{vf}$$
$$+ \sum_{v \in V^{\text{NFV}}} u_v^{(5)} \sum_{f \in F_c} n_f^{\text{CORE}} \sum_{(v_s, v_d) \in \mathcal{SD}} D_{sd}^c a_{vf}$$
$$+ \sum_{\ell \in L} \sum_{(v_s, v_d) \in \mathcal{SD}} u_\ell^{(6)} D_{sd}^c \sum_{i=1}^{n_c - 1} b_{i\ell}^\gamma$$
$$- \sum_{f \in F} \sum_{v \in V^{\text{NFV}}} u_{vf}^{(7)} a_{vf} + \sum_{f \in F} \sum_{v \in V^{\text{NFV}}} u_{vf}^{(8)} a_{vf} - u_c^{(3)}. \quad (17)$$

where RED_COST value indicates whether an optimal γ for c has been found. A non-negative value of RED_COST indicates optimality for our model.

Constraints:

$$\sum_{f \in F_c} n_f^{\text{CORE}} \sum_{(v_s, v_d) \in \mathcal{SD}} D_{sd}^c a_{vf} \leq n^{\text{CORE}} \qquad v \in V^{\text{NFV}} \quad (18)$$

$$\sum_{(v_s, v_d) \in \mathcal{SD}} D_{sd}^c \sum_{i=1}^{n_c - 1} b_\ell^{\sigma_i(c), \sigma_{i+1}(c)} \leq \text{CAP}_\ell \quad \ell \in L \quad (19)$$

$$\sum_{v \in V^{\text{NFV}}} a_{v\sigma_i(c)} = 1 \qquad i = 1, 2, \ldots, n_c \quad (20)$$

$$\sum_{\ell \in \omega^+(v)} b_{1\ell}^\gamma \geq a_{v,\sigma_1(c)} - a_{v,\sigma_2(c)} \qquad v \in V^{\text{NFV}} \quad (21)$$

$$\sum_{\ell \in \omega^+(v)} b_{1\ell}^\gamma \leq 1 - a_{v,\sigma_2(c)} \qquad v \in V^{\text{NFV}} \quad (22)$$

$$\sum_{\ell \in \omega^-(v)} b_{1\ell}^\gamma \leq 1 - a_{v,\sigma_1(c)} \qquad v \in V^{\text{NFV}} \quad (23)$$

$$\sum_{\ell \in \omega^+(v)} b_{i\ell}^\gamma - \sum_{\ell \in \omega^-(v)} b_{i\ell}^\gamma = a_{v,\sigma_i(c)} - a_{v,\sigma_{i+1}(c)}$$

$$v \in V^{\text{NFV}}, i = 1, 2, \ldots, n_c - 1 \qquad (24)$$

$$\sum_{\ell \in \omega^+(v)} b_{i\ell}^\gamma - \sum_{\ell \in \omega^-(v)} b_{i\ell}^\gamma = 0$$

$$v \in V \setminus V^{\text{NFV}}, i = 1, 2, \ldots, n_c - 1 \qquad (25)$$

$$\sum_{\ell \in \omega^-(v)} b_{(n_c-1)\ell}^\gamma \geq a_{v,\sigma_{n_c}(c)} - a_{v,\sigma_{n_c-1}(c)}$$

$$v \in V^{\text{NFV}} \qquad (26)$$

$$\sum_{\ell \in \omega^+(v)} b_{(n_c-1)\ell}^\gamma \leq 1 - a_{v,\sigma_{n_c}(c)} \qquad v \in V^{\text{NFV}}. \qquad (27)$$

Eqs. (18) and (19) are compute resource and capacity constraints, similar to those in **RMP**.

Eq. (20) ensures that each VNF of SC c is placed at least once. Eq. (21) ensures that, if $f_{\sigma_1}(c)$ is located in v, then a flow b is outgoing from v (one flow out, but no incoming flow in v). In addition, if $f_{\sigma_1}(c)$ is not located in v, then Eq. (21) is redundant. On the other hand, Eq. (23) ensures that, if $f_{\sigma_1}(c)$ is located in v, then there is no flow (b) that is incoming to v. Also, if $f_{\sigma_1}(c)$ is not located in v, then Eq. (23) is redundant.

Eqs. (26) and (27) are similar to Eqs. (23) and (21), but are related to last VNF in SC c.

Eqs. (24) and (25) are flow-conservation constraints: Eq. (24) for nodes with NFV support and Eq. (25) for other nodes.

C. Solution Scheme

The **PP_SC**(c) are solved in a round-robin fashion and the final **RMP** is solved as an ILP, as shown in [9].

V. Illustrative Numerical Examples

We run instances of our optimization model over the NSFNet topology shown in Fig. 4. Each network link has 40 Gbps capacity in each direction. Fig. 5(a) depicts one of the type of SCs deployed for Figs. 5(b), 5(c) and 5(d). Service Chain 1 is made up of Session Border Controller (SBC) and Quality of Service (QoS) VNFs which need to be traversed in sequence. Together, these two VNFs provide VoIP service with call-quality monitoring. We deploy 13 different service chains (SCs) over 20 uniformly-distributed traffic flows. All SCs comprise between two to five VNFs. In total, 33 different VNFs are used across the 13 SCs. This is a large number of distinct SCs and VNFs to be deployed. The solution for these problem instances is found in a relatively short time. The VNFs differ in functionality and CPU core requirements as given in [10].
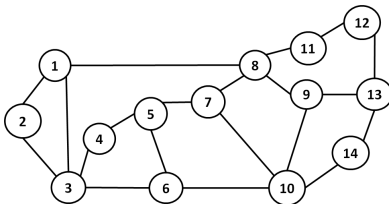

Fig. 4: NSFNet topology.

Here, we employ the following five NFV infrastructure deployment schemes and compare their network resource consumption across varying traffic and CPU cores.

- **NFV-Deg3**: In this scheme, all the degree-3 nodes in the network are made NFV-PoP. For our topology, **NFV-Deg3** will have nodes 1, 5, 6, 7, 9, and 13 as NFV-PoPs, i.e., they have NFV support.
- **NFV-Deg4**: All degree-4 nodes in the network are made NFV-PoP. In NSFNET, **NFV-Deg4** will have nodes 3, 8, and 10 as NFV-PoPs.
- **NFV-ALL**: Here, all nodes in NSFNet are NFV-PoPs.
- **NFV-SR**: Here, we are *skewed* (S) towards nodes in the *right* (R) of the network. So, nodes 9, 11, 12, 13, and 14 are made NFV-PoP. This scheme demonstrates how location of nodes is crucial for NFV-PoPs.
- **DC-X**: Here 'X' is any of the above schemes. This scheme represents a Network-enabled Cloud (NeC), as described in Section I. Here, a single DC node having massive compute resources is placed in the network and NFV-PoPs nodes are selected based on scheme 'X'. To emulate a DC, we do not have CPU core capacity constraints at DC node. In our study, we place the DC node at each of the 14 nodes of NSFNet across our model instances. After each DC placement, NFV-PoPs are selected based on 'X'. The final result is the average of results obtained from each DC placement.

As number of nodes with NFV support in the network increases, traffic flows can satisfy service demands closer to the shortest path. However, the location of NFV-PoPs is as crucial. We demonstrate this with Fig. 5(b) where **NFV-SR** and **NFV-Deg3** have more NFV-PoPs (5 and 6, respectively) than **NFV-Deg4** (3 NFV-PoPs) but still have higher network-resource consumption. This happens as central nodes (as in **NFV-Deg4**) have more traffic flows pass through them. Also, **NFV-ALL** has least network-resource consumption since all nodes in the network in this scheme are NFV-PoP. Note that we have not enforced CPU core capacity constraints here.

A NFV-PoP has limited compute resources, and VNF placement will be contingent on resources available. Fig. 5(c) shows NFV-PoPs with different CPU cores. We find that, as traffic increases, the network-resource consumption trend is the same as in Fig. 5(b) with **NFV-SR** using the most network resources while **NFV-ALL** uses the least. However, we find that infeasibility occurs for schemes at certain traffic and CPU core (n^{CORE}) values. **NFV-Deg4** becomes infeasible at $n^{\text{CORE}} = 4$ at 1 Gbps, while all schemes are infeasible for $n^{\text{CORE}} = 4$ at 2 Gbps and $n^{\text{CORE}} = 4, 8$ at 3 Gbps. This happens due to increasing CPU core demand as traffic increases. If the NFV-PoP's CPU cores are not sufficient, the scenario is not feasible. So, 2 Gbps traffic becomes infeasible at $n^{\text{CORE}} = 4$ while 1 Gbps traffic does not. Infeasibility becomes a significant challenge with increasing traffic.

A massive compute resource node (e.g., a Data Center (DC)) in the network can avoid infeasibility as traffic can be routed to DC when NFV-PoPs have insufficient compute resources. Fig.
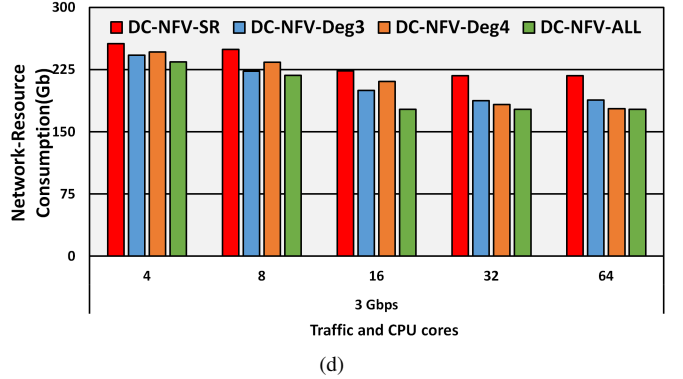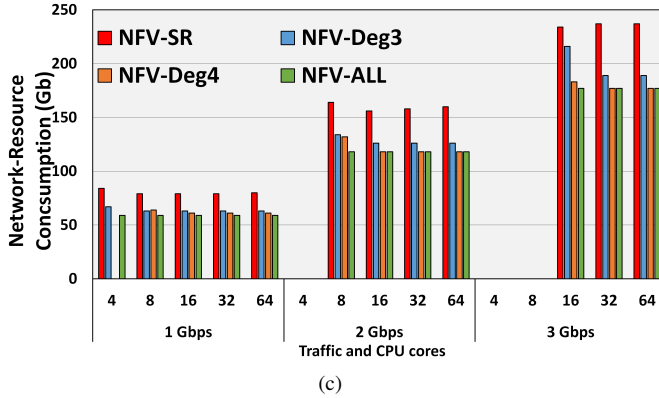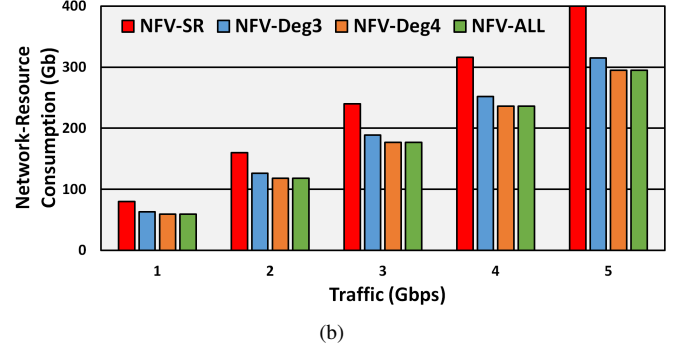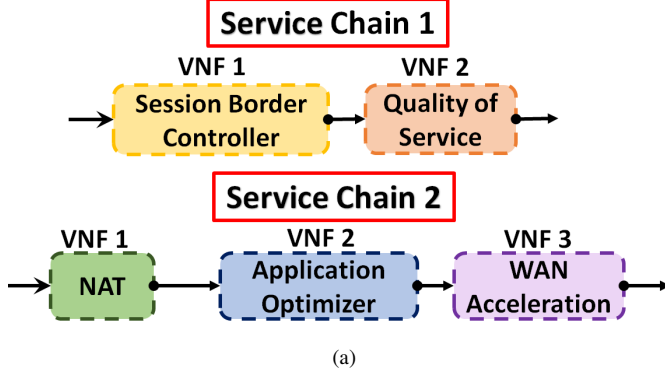
Fig. 5: Example of service chains deployed and results.

5(d) shows that **DC-X** scheme's make infeasible n^{CORE} values of Fig. 5(c) become feasible. At $n^{\text{CORE}} = 4$ and 3 Gbps, all NFV-PoP schemes were infeasible in Fig. 5(c). When we add a DC node to each scheme, we find that they become feasible for $n^{\text{CORE}} = 4$ and 3 Gbps in Fig. 5(d). Similar observation is made for $n^{\text{CORE}} = 8$. This becomes possible as the traffic can be redirected to DC when compute resources are scarce.

Our discussion on Figs. 5(b), 5(c), and 5(d) makes the case for a Network-enabled Cloud (NeC) for Network Function Virtualization Infrastructure (NFVI). A NeC will avoid resource constraints encountered in a NFVI having NFV-PoPs exclusively. Also, NeC can avoid congestion arising from a centralized NFVI (like a DC) since NFV-PoPs can share traffic load which makes NeC the ideal NFV Infrastructure (NFVI).

VI. CONCLUSION

We introduced the problem of service-chain placement and how, with Network Function Virtualization (NFV), we can use Virtual Network Functions (VNFs) for more flexible/agile service chaining. We developed a decomposition model for placement of multiple VNF service chains (SCs) and routing of traffic flows. Our model helps in solving this complex problem in a relatively small amount of time. Our objective was to demonstrate the reduction in network-resource consumption if we have more NFV Points of Presence (NFV-PoPs). We

further showed that a Network-enabled Cloud (NeC) would be ideal for NFV Infrastructure (NFVI) since it avoids the resource constraints of a NFVI with NFV-PoPs only. NeC also avoids congestion problem of centralized NFVI (e.g., Data Center) since it has NFV-PoPs to share the service traffic load.

REFERENCES

[1] ETSI, "Network functions virtualisation: Introductory white paper." portal.etsi.org/NFV/NFV_White_Paper.pdf, 2012.

[2] Ericsson, "The real-time cloud - combining cloud, NFV and service provider SDN." www.ericsson.com/res/docs/whitepapers/wp-sdn-and-cloud.pdf, 2014.

[3] IETF, "Network service chaining problem statement." https://tools.ietf.org/html/draft-quinn-nsc-problem-statement-00, 2013.

[4] H. Moens and F. De Turck, "VNF-P: A model for efficient placement of virtualized network functions," in *Proc. 10th International Conference on Network and Service Management (CNSM)*, pp. 418–423, Nov. 2014.

[5] B. Addis, D. Belabed, M. Bouet, and S. Secci, "Virtual network functions placement and routing optimization," *https://hal.inria.fr/hal-01170042/*, 2015.

[6] M. C. Luizelli, L. R. Bays, L. S. Buriol, M. P. Barcellos, and L. P. Gaspary, "Piecing together the NFV provisioning puzzle: Efficient placement and chaining of virtual network functions," in *Proc. IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 98–106, May 2015.

[7] M. Bari, S. Chowdhury, R. Ahmed, and R. Boutaba, "On orchestrating virtual network functions in NFV," *Computing Research Repository*, vol. abs/1503.06377, 2015.

[8] A. Gupta, M. Habib, P. Chowdhury, M. Tornatore, and B. Mukherjee, "Joint Virtual Network Function Placement and Routing of Traffic in Operator Networks," *Technical Report, UC Davis*, 2015.

[9] B. Jaumard, C. Meyer, and B. Thiongane, "On column generation formulations for the RWA problem," *Discrete Applied Mathematics*, vol. 157, pp. 1291–1308, 2009.

[10] Cisco, "Cisco Cloud Services Router 1000V 3.14 Series Data Sheet." http://www.cisco.com/c/en/us/products/collateral/routers/cloud-services-router-1000v-series/datasheet-c78-733443.pdf, 2015.