Scalable Solution for Service Function Chaining Embedding in Hybrid Optical-Electronic Networks

Rongping Lin, Yangzhou Wu, Shan Luo, Bill Moran and Moshe Zukerman, *Life Fellow, IEEE*

***Abstract*—Network function virtualization (NFV) leverages software and general-purpose computation devices to provide various network services through service function chaining (SFC), where data flows through a sequence of network functions to obtain the desired services. In this context, the growth of network traffic due to the proliferation of network services presents a challenge. To address this, hybrid optical-electronic networks, combining the advantages of both optical and electronic networks, have been applied to create a more efficient and flexible communication system which also transport traffic through the network functions of SFC. In this paper, we address the problem of embedding SFC in a hybrid optical-electronic network. This problem is even more challenging than the SFC embedding problem in an electronic network, which is already known to be NP-hard. To tackle this problem, we propose a scalable SFC embedding algorithm in hybrid optical-electronic networks that optimizes both optical and electronic bandwidth allocations as well as computation resource allocation. We first provide an integer linear programming formulation for this optimization problem and then propose a scalable heuristic algorithm based on the randomized rounding method to achieve near-optimal solutions. Our numerical results demonstrate that the proposed algorithm achieves both guaranteed performance and scalability through efficient running time.**

***Index Terms*—Service function chaining, hybrid optical-electronic networks, randomized rounding**

I. Introduction

NETWORK service providers are investing significant resources in new network infrastructures to meet the demands of rapidly increasing numbers of smart devices. Traditional network middleboxes, such as firewalls (FW), intrusion detection systems (IDS), deep packet inspection (DPI), etc., are used to provide various network functions, but these dedicated hardware devices are costly and time-consuming to deploy. To address these challenges, network function virtualization (NFV) technology [1], [2] was proposed to remove the reliance of network functions on dedicated hardware. This decoupling of network functions from dedicated hardware devices uses software and standard computational devices to provide virtual network functions (VNFs), which can be deployed on standard computation servers, making the network much more flexible. This allows relocation of VNFs or instantiation of new VNFs without the need for additional dedicated hardware devices when traffic demand changes, reducing network operating costs and accelerating the development of new network services. To provide a new network service, then, a new service function chain (SFC) [3] is introduced, composed of an ordered set of VNFs, so that the data flow along the SFC through these VNFs in the specific order. However, embedding an SFC into a physical network and connecting the VNF instances in their corresponding order requires both computation and bandwidth resources [4], and this problem has been thoroughly studied. The optimal selection of SFC embedding is an NP-hard combinatorial optimization problem [5].

With the increasing adoption of NFV technology to support different network services [6], [7], the underlying physical network is expected to accommodate diverse bandwidth transmission requirements, necessitating the provision of a variety of bandwidth services to cater for the varying needs of those services. Electronic networks provide fine bandwidth granularities and are typically efficient at a large number of data connections that have various low bandwidth requirements, but it is costly to handle high-bandwidth traffic due to high costs of the high speed switches and high energy consumption. Optical networks, on the other hand, are well-equipped to provide a limited number of high-bandwidth connections, though they may have limitations in offering fine-grained bandwidth connections. A hybrid optical-electronic network contains both electronic transmission channels and optical transmission channels for high efficiencies, where high-bandwidth connections (usually there is only a small number of) are transmitted by all-optical transmission channels, and low-bandwidth connections (there is a large number of) are transmitted by electronic transmission channels.

To fulfill the diverse bandwidth transmission requirements from NFV networks, this paper considers the use of hybrid optical-electronic networks that contains optical network and electronic network simultaneously, and can efficiently allocate bandwidth for different connections to meet their bandwidth requirements. There are layered graphs [8] that have been employed to represent the IP-over-WDM networks contain electronic layer and optical layer, in which the electronic layer relies on the optical layer's services that transmits data in the optical domain, and data switching occurs in electronic domain. This approach allows for consistent data transmission across various connections with differing bandwidth require-

This work was supported by a grant from the Sichuan Science and Technology program (2023YFG0298), and by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 11200721). (Corresponding authors: Rongping Lin and Shan Luo)

R. Lin and Y. Wu are with the School of Information and Communication Engineering, University of Electronic Science and Technology of China (UESTC), China (e-mail: linrp@uestc.edu.cn, wu_yz0@163.com).

S. Luo is with the School of Aeronautics and Astronautics, UESTC, China (e-mail: luoshan@uestc.edu.cn).

B. Moran is with the Department of Electrical and Electronic Engineering, University of Melbourne, Melbourne, VIC 3010, Australia (e-mail: wmoran@unimelb.edu.au).

M. Zukerman is with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong SAR (e-mail: m.zu@cityu.edu.hk).

ments, necessitating substantial data conversion between optical and electronic formats. The primary distinction between the IP-over-WDM network and the hybrid network lies in the presence of two distinct physical layers within the hybrid network: the electronic layer and the optical layer. These layers can independently and concurrently provide data transmissions. The hybrid network selects the appropriate transmission method, whether electronic or optical, based on the specific connection requirements. More specifically, connections with lower bandwidth demands opt for electronic transmissions to leverage the flexibility and fine-grained nature of electronic processing. On the other hand, connections with larger bandwidth requirements utilize optical transmissions due to the high efficiency of optical transmission for handling substantial data volumes. In the context of this paper, the connections being considered are assumed to be heterogeneous, with significantly different bandwidth requirements. This justifies the adoption of the hybrid optical-electronic network, as it allows for efficient and tailored data transmission across this diverse range of connections. It is worth noting that this work on SFC embedding in hybrid optical-electronic networks has potential applications in LANs. LANs offer the advantage of easy deployment for both electronic and optical networks, and where SFC requests are centrally processed, similar to data center networks.

To successfully embed an SFC request in a network, the VNFs of the SFC must be embedded to physical network nodes (allocating computation resources) in a specific order to ensure proper user data processing. This is known as VNF embedding. Additionally, data connections between adjacent VNFs need to be established by allocating appropriate bandwidth resources along the derived paths, referred to as virtual link embedding. It is important to note that VNF embedding and virtual link embedding are interconnected problems, and the challenge lies in optimally embedding VNFs and simultaneously setting up paths for virtual links. Embedding an SFC optimally within an electronic network is known to be an NP-hard problem [9]. However, the challenge becomes even more complex when considering SFC embedding optimization in a hybrid optical-electronic network. In comparison to an electronic network, a hybrid optical-electronic network involves additional constraints related to the optical network, such as wavelength continuity and the interaction between electronic and optical networks. These constraints, combined with the existing constraints of the electronic network and computation resources, further complicate the optimization process. Consequently, the challenge becomes more difficult due to the tight coupling of three key factors: VNF embedding, electronic bandwidth allocation, and optical bandwidth allocation, all of which collectively determine the performance of the SFC embedding.

To address this challenge, in this paper, we provide a scalable algorithm for SFC embedding for hybrid optical-electronic networks. Our algorithm not only improves time efficiency but also guarantees near-optimal allocation of bandwidth and computation resources. The main contributions of this paper are as follows.

1) Provision of an integer linear programming (ILP) formulation for the SFC embedding problem in hybrid optical-electronic networks, aiming to maximize operator revenue. The formulation encompasses joint optimization of VNF embedding and virtual link embedding.
2) In recognition of the NP-hardness of the SFC embedding problem in hybrid optical-electronic networks, we propose an approximate scalable heuristic algorithm based on the randomized rounding method. This algorithm efficiently addresses the problem by employing an auxiliary graph to solve VNF embedding and virtual link embedding. The problem is transformed into a path routing problem within the auxiliary graph. We also provide a statistical analysis of the algorithm performance.
3) Extensive numerical results demonstrate the following: (1) for small-sized problems, our heuristic algorithms produce solutions that closely approximate those obtained from the ILP, but with significantly shorter running times; (2) our algorithms provide solutions within three seconds for various problem sizes, which cannot be achieved by the ILP within a reasonable time; (3) our heuristic algorithms exhibit efficient performances in a larger randomly generated network comprising 40 nodes and 100 links.

The remainder of this paper is organized as follows. Section II outlines the related work. Section III presents the discussion of the SFC embedding in hybrid optical-electronic networks, as well as an ILP formulation of the problem. The design details of the SFC embedding algorithm based on the randomized rounding method are provided in Section IV. In Section V, we present and analyze the numerical results. Finally, this paper is concluded in Section VI.

II. Related Work

Network service deployment with SFC embedding in NFV networks is flexible, allowing resource utilization of the physical network to be optimized by the allocation of underlying existing resources. Many SFC embedding problems become scheduling and decision-making optimization problems: the goal being to accommodate as many requests as possible with limited physical network resources. The joint optimization problem of SFC embedding *and* network resource allocation was discussed in [10]. There the column generation method was applied to derive the solution while minimizing service completion time. Fan *et al.* [11] investigated an online dynamic SFC embedding method under reliability constraints. The usage of network resources was minimized to provide a joint protection mechanism, which utilizes fewer network resources than the traditional protection mechanism. Bouten *et al.* [12] considered different multimedia traffic and provided a mathematical model for joint optimization of data center resource dimensioning and SFC resource consumption. In [13], an integer linear programming model for the SFC embedding problem was provided to minimize the cost, including VNF deployment cost, energy consumption cost and traffic forwarding cost, while taking account of the time delay of user demands. A heuristic algorithm based on dynamic programming was designed to solve larger-sized problems. In

[14], the SFC embedding problem was investigated for multi-domain networks, where the end-to-end delay requirement was taken into account, and cross-domain data traffic was reduced to minimize deployment costs. In [15], the dynamic SFC embedding problem was investigated for data center scenarios, with minimization of the embedding cost. An efficient online algorithm involving the dividing of the original problem into several subproblems was proposed. Zheng et al. [16] proposed a logarithm approximate algorithm to minimize the cost of SFC embedding, which has provable bounds. Zheng et al. [17] also introduced network function parallelism method that allows multiple SFs to run in parallel to reduce the SFC processing delay of the serial VNFs. Laroui et al. [18] proposed a SFC embedding algorithm in the virtual mobile edge computing environment over Internet of Things (IoT) networks, where the optimal virtual mobile edge are selected to embed VNFs of the SFC, and heterogeneous computing resources at virtual mobile edges are considered. Liu et. al. [19] investigated the SFC embedding problem with deep reinforcement learning, and the dynamic network state including computation resources and bandwidth resources was considered, which is a perfect scenario that a reinforcement learning can tackle. Zheng et. al. [20] investigated the fault-tolerance of SFC embedding in edge networks, where multiple concurrent heterogeneous failures, like physical/virtual network failures and hardware/software failures, were considered during the SFC embedding. These studies considered many aspects of the SFC embedding problem, but the connection services provided by the substrate network are not investigated, which may degrade the system performance if SFC requests have quite different connection requirements.

As traffic demand for the bandwidth resources of the network increases rapidly, an optical network that can provide sufficient bandwidth and low power consumption is being introduced into NFV networks. There is some literature on SFC embedding in optical networks. In [21], the deployment and instantiation of VNF in an optical network were investigated, and a mixed integer linear programming (MILP) formulation of the problem was provided. Physical network resource allocation of VNF embedding and routing was designed via a two-player game to achieve optimal utilization of network resources. In [22], a new network architecture that can process high-volume SFC in a data center optical network was designed, where the aggregated super-large data flow was transferred to the high-capacity VNF node in the optical domain with a ROADM (reconfigurable optical add-drop multiplexer) to reduce energy consumption significantly. There are also publications on SFC embedding in elastic optical networks. Fang *et al.* [23] focused on resource allocation for SFC embedding in elastic optical networks across data centers. They proposed a heuristic algorithm based on the longest common subchain algorithm to efficiently reuse VNF nodes, using a load-balancing algorithm to optimize the usage of spectrum resources. Zeng *et al.* [24] studied the multicast NFV tree embedding problem in an elastic optical network. They provided a formulation of the joint optimization problem of VNF embedding, link spectrum allocation, and multicast routing with the objective function of minimizing the network cost (VNF deployment cost and spectrum usage). Zhang et. al. [25] investigated the SFC embedding problem in a space-division multiplexing elastic optical network, where the frequency slot allocation is the key factor in the virtual link embedding. Yu et. al. [26] approximated the SFC embedding problem with the priority sorting that gives the specific order of the VNF and Virtual link embedding to largely reduce the complexity of the problem in elastic optical networks. Wang et. al. [27] investigated the SFC embedding problem in elastic optical networks, where the allocations of computation resources and spectrum resources have to be coordinated for the performance efficiency. All these studies focus on the optical network, and take constraints of the optical network into considerations during the SFC embeddings. However, in this paper, the hybrid optical-electronic network contains optical and electronic networks simultaneously, and the bandwidth resource allocation should be optimized in the network that combines these two different networks. Lin *et al.* [28] considered SFC embedding in hybrid optical-electronic networks, where an approach based on Markov chain modeling was used to instantiate VNFs and set up/tear down optical paths that enable the SFC requests to be embedded with a maximal profit-based utility function.

To the best of our knowledge, it is still an open research on the scalable solution for the SFC embedding problem in hybrid optical-electronic networks, while taking into account the joint optimization of the allocation of both the optical and electronic bandwidths as well as the computation resources. To address this research problem, this paper proposes a randomized rounding-based scalable algorithm to solve the SFC embedding problem in hybrid optical-electronic networks while optimizing the optical and electronic bandwidths and computation resource allocations.

III. Preliminaries and Problem Formulation

In this section, we first introduce and illustrate, by an example, SFC embedding in a hybrid optical-electronic network. Then the ILP formulation of the SFC embedding problem for a set of SFC requests is provided.

A. Hybrid Optical-Electronic Networks

In a hybrid optical-electronic network, servers are deployed at physical network nodes; these can be used to process network service functions and forward and route data traffic. In this work, each physical node may simultaneously provide different types of network service functions. This practical assumption makes the SFC embedding problem more complicated because the search space of the combination optimization becomes significantly larger.

We give an SFC embedding example in a hybrid optical-electronic network, and we utilize a layered graph to illustrate the concept of the hybrid optical-electronic network as depicted in Fig. 1. In this example, the network has four layers. The top layer shows that the SFC request containing a set of VNFs be embedded in the network and sequentially connected. The second layer concerns network topology; each network node contains an electronic router and an optical

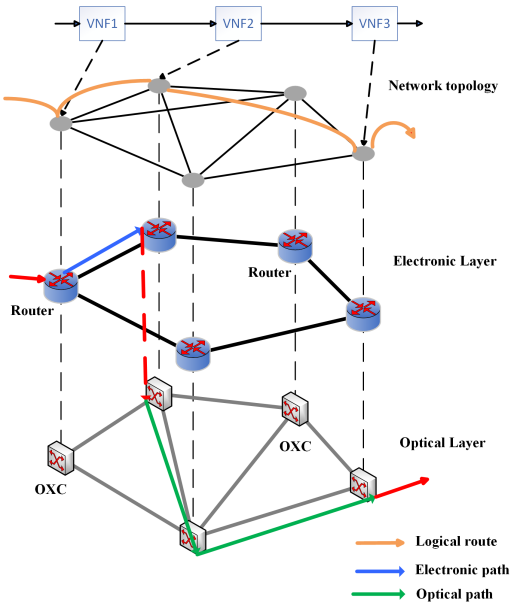


Fig. 1: SFC embedding in a hybrid optical-electronic network

cross-connect (OXC) shown at the third and the bottom layer, respectively. Each network node supports both electronic packet switching and optical switching. Computational servers that sustain VNFs are also included in network nodes. The links of the network topology layer represent the links of the hybrid optical-electronic network, each of which can be either an electronic link or an optical link. A link in the network topology indicates a communication link of some kind between two network nodes that are, thereby, adjacent. In this example, three different VNFs are embedded at three different network nodes, and the connections between VNFs are different: from VNF1 to VNF2, an electronic path is used, and from VNF2 to VNF3, an optical path. The reason for the different paths may be that VNF2 may generate considerable new traffic after processing, so an optical path is needed for traffic transmission from VNF2 to VNF3.

B. Problem Formulation

We consider the problem of optimizing static SFC embedding in a hybrid optical-electronic network. Given a set of SFC requests, we seek to allocate network and computing resources to maximize the revenues of SFC embedding while taking into account constraints such as the computing resources of network nodes, limitations on electronic/optical link bandwidth, and other factors such as the continuity of wavelengths on optical paths. SFC embedding involves two main processes: VNF embedding and virtual link embedding. VNF embedding entails selecting network nodes that match the network function types and have sufficient available computing resources for each VNF of the SFC. Virtual link embedding enables the transfer of the traffic flow of the SFC through the selected network nodes based on the sequence of VNFs, using connections such as electronic paths or optical paths. The notations used in the paper are defined in Table I. The details of the ILP formulation are as follows.

TABLE I: Summary of notations

Notation	Description
N^s	Set of given physical nodes
L^s	Set of physical links including optical and electronic links
$G^s(N^s, L^s)$	Directed graph represents the substrate network
F_r	Set of VNFs in request r
L_r	Set of virtual links in request r
$r(F_r, L_r, p, q)$	An SFC request r with ingress node p and egress node q
R	Set of all SFC requests
S_n	Set of network function types that physical node n can provide
$\tau(i)$	Network function type of the VNF node i
C_n^s	Available computation resource of network function s at physical node n
$C_{mn,w}$	Available bandwidth of wavelength w in physical link mn
F	Number of wavelengths in an optical link
$C_{mn,0}$	Available bandwidth of electronic links in physical link mn
d_i^r	Computation requirement of the VNF node i in SFC request r
d_{ij}^r	Bandwidth requirement of virtual link ij in SFC request r
p_r	Revenue of embedding the SFC request r
G_r^{Aux}	Auxiliary graph for SFC request r
q_r^l	Cost of the lth embedding path for the SFC request r
x_r	Boolean decision variable. It takes the value 1 when the SFC request r is embedded successfully, otherwise 0
$A_{k,w}^{i,r}$	Boolean decision variable. It takes the value 1 if the VNF node i of SFC request r is embedded to physical node k, and the virtual link between the VNF node i to the next VNF node uses the optical path with wavelength w ($w = 0$ means using an electronic link), otherwise 0
$B_{k,w}^{i,r}$	Boolean decision variable. It takes the value of 1 if the VNF node i of SFC request r is embedded to physical node k, and the virtual link between the VNF node i to the previous VNF node uses the optical path with wavelength w ($w = 0$ means using an electronic link), otherwise 0
$M_{mn,w}^{ij,r}$	Boolean decision variable. It takes the value of 1 if virtual link ij of SFC request r traverses physical link mn with wavelength w ($w = 0$ means using electronic link), otherwise 0

$$Maximize: \sum_{r \in R} x_r p_r \tag{1}$$

subject to the following constraints.

$$\sum_{k \in N^s} \sum_{w \in [0,F]} A_{k,w}^{i,r} \leq 1 \qquad \forall i \in F_r, r \in R \tag{2}$$

$$\sum_{k \in N^s} \sum_{w \in [0,\ F]} A_{k,w}^{1,r} = x_r \qquad \forall r \in R \tag{3}$$

$$\begin{aligned} \sum_{m \in N^s} M_{mk,w}^{ij,r} - \sum_{n \in N^s} M_{kn,w}^{ij,r} = B_{k,w}^{j,r} - A_{k,w}^{i,r} \\ \forall ij \in L_r, r \in R, k \in N^s, w \in [0,F] \end{aligned} \tag{4}$$

$$\sum_{w \in [0,F]} A_{k,w}^{i,r} = \sum_{w \in [0,F]} B_{k,w}^{i,r} \quad \forall i \in F_r, r \in R, k \in N^s \tag{5}$$

$$\sum_{r \in R} \sum_{i \in F_r | \tau(i) = s} \sum_{w \in [0,F]} d_i^r A_{n,w}^{i,r} \leq C_n^s \quad \forall n \in N^s, s \in S_n \tag{6}$$

$$\sum_{r \in R} \sum_{ij \in L_r} d_{ij}^r M_{mn,w}^{ij,r} \leq C_{mn,w} \quad \forall mn \in L^s, w \in [1,F] \tag{7}$$

$$\sum_{r \in R} \sum_{ij \in L_r} d_{ij}^r M_{mn,0}^{ij,r} \leq C_{mn,0} \quad \forall mn \in L^s \tag{8}$$

$$x_r \in \{0,1\} \quad \forall r \in R \tag{9}$$

$$A_{k,w}^{i,r} \in \{0,1\} \quad \forall i \in F_r, r \in R, k \in N^s, w \in [0,F] \tag{10}$$

$$B_{k,w}^{i,r} \in \{0,1\} \quad \forall i \in F_r, r \in R, k \in N^s, w \in [0,F] \tag{11}$$

$$M_{mn,w}^{ij,r} \in \{0,1\} \quad \forall ij \in L_r, r \in R, mn \in L^s, w \in [0,F] \tag{12}$$

Equation (1) describes the optimization objective of the problem, namely, to maximize the revenue of embedding the batch of SFCs. If an SFC request is embedded successfully, the network operator obtains the corresponding revenue. In this problem, some SFC requests may not be embedded successfully because of the constraints on network and computation resources. In the objective function, $x_r p_r$ represents the revenue of embedding SFC request r if request r is embedded (and it is equal to zero otherwise). There are many ways to set the revenue p_r for embedding an SFC request r. Here, for simplicity, the revenue from an accepted SFC request r is set to be equal to the sum of the total computation requirements $\sum_{i \in F_r} d_i^r$ and the total bandwidth requirements $\sum_{ij \in L_r} d_{ij}^r$ of the SFC request r. In this way, the revenue of embedding an SFC increases with resource usage of the SFC, and the calculation of p_r becomes possible once the SFC is known.

The constraints of our optimization problem are given by Equations (2)–(12). Constraint (2) ensures that each VNF is embedded in, at most, one physical node. This implies that a VNF cannot be split up into multiple physical nodes for embedding. Constraint (3) provides the connection between the two sets of decision variables x_r and $A_{k,w}^{1,r}$. If the first VNF of an SFC is embedded into a physical node, the right hand of the equation should equal 1, and this forces x_r to take the value 1. Constraint (4) guarantees the virtual link embedding in the electronic network or optical network and ensures flow conservation at each physical node. There are three cases according to the value of the right side of the equation: 1) the right side of the equation equals 0 ($A_{k,w}^{i,r}$ and $B_{k,w}^{j,r}$ have the same value) — this implies that the number of incoming flows and outgoing flows that traverse node k are equal. In case 2) the right side of the equation equals 1 ($B_{k,w}^{j,r}$ equals 1 and $A_{k,w}^{i,r}$ equals 0), and this covers the situation where the VNF node j is embedded in physical node k but VNF node i is not. Finally, in case 3) the right side of the equation is equal to -1 ($B_{k,w}^{j,r}$ equals 0 and $A_{k,w}^{i,r}$ equals 1); here, the VNF node i is embedded in the physical node k and the VNF node j is not. By introducing variables $A_{k,w}^{i,r}$ and $B_{k,w}^{j,r}$ and adding the subscript w to represent the optical wavelength, we ensure the wavelength continuity constraint when using optical links. Constraint (5) ensures that the two variables $A_{k,w}^{i,r}$ and $B_{k,w}^{i,r}$ indicating the embedding of the VNF node i have consistent values, where the two sets of variables indicate the VNF node i is embedded in the same physical node. Constraint (6) concerns the computation resource: the total usage of the computation resource has to be no larger than the computation resource available. Constraint (7) concerns the bandwidth resource of an optical link; that is, it forces the bandwidth of optical links used by SFC requests not to exceed the available bandwidth of physical links. Constraint (8) is a similar bandwidth constraint for an electronic link. Constraints (9)–(12) give the value ranges of Boolean variables of the ILP formulation.

By considering bandwidth and computation limitations, we assume that we are already committed to all possible costs including all capital and operational expenditures associated with these resources and now our aim is to maximize the revenue subject to these resource limitations. For example, we do not consider costs of optical-electronic-optical (OEO) amplifiers because they are part of the already committed bandwidth costs. For every link regardless of its length, we consider given bandwidth limitation. Clearly, for long cables. OEO amplifiers will be required and they are part of the available resources. In particular, OEO amplifiers are not considered as switches but are considered as part of the available bandwidth resources. Meanwhile, in the problem of this paper, the virtual link embedding process establishes either an electronic path or an all-optical path for transmitting data between two adjacent VNFs. It should be emphasized that the endpoints of the embedding path house the VNFs responsible for processing data in the electronic domain. If an all-optical path is employed for data transmission between two VNFs, the data is initially converted into an optical signal at the source node, facilitating optical transmission along the path. Upon reaching the destination node, the data is then converted back into the electronic domain for further processing by the VNF. Importantly, it is worth noting that in our model, there is no OEO conversion during the optical data transmission along the path and this assumption, as mentioned above does not exclude OEO amplifiers that their costs are already included in the available limited bandwidth.

This formulation addresses the problem of maximizing revenue by accommodating a batch of SFC requests with limited network resources. The problem is static, meaning that the SFC requests are given in advance, and there is no dynamic arrival or departure of SFC requests in the network. However, it is important to note that the methods proposed in this paper can be adapted for dynamic scenarios as well. In a dynamic scenario, where SFC requests can arrive and leave the network dynamically, the methods presented in this paper can still be applied. If the immediate service response for an SFC request is not required, we can wait for a group of requests to accumulate and then utilize the methods provided by this paper to embed them efficiently. This process can be repeated as subsequent batches of requests arrive and are embedded effectively. This approach can even be extended to handle requests that arrive one by one, treating a single request as a batch of size one. However, it is important to clarify that while the methods proposed in this paper have the potential to address realistic scenarios involving dynamic SFC request arrivals, our focus is primarily on the static scenarios. We have not specifically considered dynamic scenarios in this study.

IV. Algorithm Design

In this section, we describe our proposed SFC embedding algorithm based on randomized rounding and analyze its performance. The above ILP problem for the embedding of batch SFC requests is NP-hard, and so cannot be solved in polynomial time. We propose, therefore, a time-efficient heuristic algorithm based on the randomized rounding method to find a scalable and near-optimal solution for the ILP problem.

The randomized rounding method [29] takes the solution to the relaxed LP transform of the ILP problem and make randomized rounding for the decision variables to provide a nearly optimal solution of the ILP. Accordingly, in this paper, the ILP problem is first relaxed to an LP problem by transforming the integer decision variables into continuous variables. This transformation reduces the difficulty: the optimal solution to the LP problem can be obtained in polynomial time. Then, the optimal solution of the LP problem is used to construct an integer (near-optimal) feasible solution of the original ILP problem.

A. Auxiliary Graph Construction Algorithm

As discussed above, the optimal solution to the relaxed problem is used to derive embedding solutions for SFC requests. The values of the variables in the optimal solution of the LP problem may not be integers, and an auxiliary graph is provided to convert these values as link costs in the graph. In this way, the joint optimization problem of VNF and virtual link embeddings is converted to a constrained path routing problem in the graph.

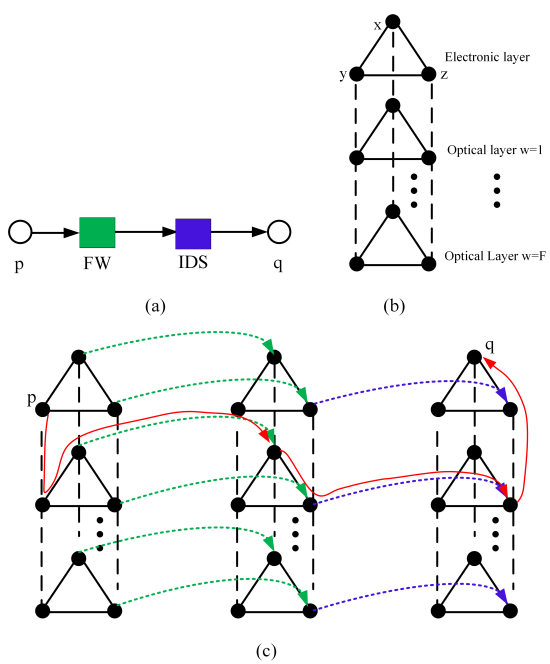


Fig. 2: Auxiliary graph design

Fig. 2 illustrates the design of an auxiliary graph. Fig. 2(a) shows the SFC request that contains two different VNFs (FW and IDS) and ingress and egress nodes. Fig. 2(b) shows the hybrid optical-electronic network, where the first layer is the electronic layer, and the other layers are optical layers operating on different wavelengths. Fig. 2(c) shows the auxiliary graph design for the SFC request based on the hybrid optical-electronic network, where the hybrid network is duplicated three times, and links are added to connect them. We note that the added links between copies of the hybrid network are used to represent the embedding of SFCs into the specific physical nodes of the link ends if the routing path traverses the link. For example, the SFC request contains two VNFs, FW and IDS, and in the auxiliary graph, there are three copies of the hybrid network. The links between the first and the second copies denote the FW embedding at the specific end nodes of the link. Links between the second and the third copies denote the IDS embedding at the specific end nodes of the link. The embedding solution for the SFC request is shown in Fig. 2(c) as a red line routing path from ingress node p to egress node q, where the FW is embedded into the physical node x because the link from physical node x is used between the first and the second copies, and the IDS is embedded into the physical node z because the link from node z is used between the second and the third network copies. Meanwhile, the three virtual links are embedded into optical paths from node y to x with wavelength $w = 1$, the optical path from node x to z with wavelength $w = 1$, and the electronic path from node z to x in order.

Algorithm 1 shows the details of building an auxiliary graph for the SFC request r. First, $|F_r| + 1$ copies (only physical nodes) of the hybrid network are added to the auxiliary graph. The optimal solution of the LP problem contains continuous values of decision variables, representing the usage of network resources (bandwidth and computation), and can be exactly map to costs of links in the auxiliary graph. Then, the auxiliary graph contains these continuous values and the final embedding solution is derived, based on them.

In Algorithm 1, from line 2, links of the auxiliary graph are added, and link costs are set according to the optimal solution of the LP problem for the SFC request r. From line 2 to line 4, links are added in each layer, comprising hybrid network copies in the auxiliary graph to represent the results of the virtual link embedding. Specifically, if the value of the variable $M^{ij,r}_{mn,w}$ is non-zero in the optimal solution of the LP problem, link $m \to n$ is added to the corresponding layer, and the link cost is set to be $M^{ij,r}_{mn,w}$. Similarly, from line 5 to line 7, if the variable $A^{i,r}_{k,w}$ has a non-zero value in the optimal solution, a link between the two physical nodes (the same node k) at adjacent network copies for that VNF embedding is added at network layer w, and the link cost is set to be $A^{i,r}_{k,w}$. After the construction of the auxiliary graph, there are multiple paths from the ingress to the egress representing multiple embedding selections of the request.

B. Embedding Path Generation Algorithm

After the auxiliary graph G^{Aux}_r is constructed for SFC request r, all the variable values $A^{i,r}_{k,w}$ and $M^{ij,r}_{mn,w}$ for SFC request r in the optimal solution of the LP problem are attached to the auxiliary graph, as described. The next stage

Algorithm 1: Auxiliary Graph Construction

Input: Current physical network $G^s(N^s, L^s)$, optimal solution of the LP problem for request r $Opt_{LP} : \{A^{i,r}_{k,w}, M^{ij,r}_{mn,w}\}$
Output: Auxiliary graph G^{Aux}_r for SFC request r
//Add nodes
Add $|F_r| + 1$ copies of the hybrid network (only network nodes) into the auxiliary graph
//Add links
for $ij \in L_r$, $mn \in L^s$,$w \in [0, F]$ **do**
  if $M^{ij,r}_{mn,w} > 0$ **then**
    Add link $m \to n$ in layer w of the network copy for virtual link ij, and set the link cost to be $M^{ij,r}_{mn,w}$
for $i \in F_r$,$k \in N^s$,$w \in [0, F]$ **do**
  if $A^{i,r}_{k,w} \neq 0$ **then**
    Add a link between two physical nodes (the same node k) at adjacent network copies for VNF i embedding and set the link cost to be $A^{i,r}_{k,w}$
Add links (vertical dash lines) with zero cost for the same physical nodes between layers in each network copy.

is to find all embedding paths (from ingress node p to egress node q) on the auxiliary graph for SFC request r and save them in Γ_r.

Algorithm 2 provides the details of finding these embedding paths in the auxiliary graph. In line 1, a path from ingress p to egress q is derived by a depth-first search. According to the auxiliary graph design, this path provides an embedding solution for the SFC request r, except that the Boolean variables may have continuous values, and multiple paths may provide different embedding selections. However, some of these embedding paths may be infeasible for the original ILP problem. In line 2, the cost of the path is set as the minimal link cost in the path. In line 3, this path is saved in the embedding path set Γ_r. The cost of this path is used for the subsequent randomized rounding to choose embedding paths in the final embedding solution. Finally, the link costs in the auxiliary graph are updated. Specifically, the link cost along the path ph is reduced by the cost of the path q^l_r, and links with a cost of 0 are deleted.

The above operations can be understood as stripping out the embedding paths from the auxiliary graph one by one until there is no path from the ingress to the egress; as a result, all embedding paths are extracted from the auxiliary graph.

C. Randomized Rounding based SFC Embedding Algorithm

The randomized rounding-based SFC embedding algorithm is provided in Algorithm 3. In line 1, the ILP problem is relaxed to the LP problem and the optimal solution Opt_{LP} of the LP problem is obtained in line 2. This solution cannot be directly used as an embedding solution for the original ILP problem: we transform the LP optimal solution into a

Algorithm 2: Embedding Path Generation

Input: Auxiliary graph G^{Aux}_r for SFC request r
Output: Embedding path set Γ_r
1 **while** *There is a path ph from p to q* **do**
  //Assign the cost to the path
2 q^l_r = the minimal link cost in ph,
3 $\Gamma_r = \Gamma_r \cup ph$
  //Update link costs in the auxiliary graph
4 Reduce link costs by q^l_r along path ph
5 If a link cost is equal to 0, delete this link from G^{Aux}_r
6 $l = l + 1$

feasible integer (near-optimal) solution for the ILP problem using randomized rounding.

From line 4 to line 10, an embedding solution is derived for each SFC request. In line 4, Algorithm 1 is called to build an auxiliary graph G^{Aux}_r for SFC request r, based on the hybrid network and the LP optimal solution Opt_{LP}. Then, the SFC embedding problem in the hybrid network is converted to the problem of finding paths from the ingress node to the egress node in the auxiliary graph followed by the selection of a proper path as the final embedding solution. Then Algorithm 2 is called to find all the embedding paths Γ_r in the auxiliary graph, where an embedding path corresponds to an embedding solution of SFC request r.

According to the solution of the LP problem, the cost sum $\sum_{l \in \Gamma_r} q^l_r$ of embedding paths in Γ_r may be less than one. This implies that there is a probability of $1 - \sum_{l \in \Gamma_r} q^l_r$ that this SFC request is blocked. The selection of an embedding path as the final embedding solution is decided by the embedding path cost q^l_r. From line 7 to line 10, an embedding path is selected from the set Γ_r as the embedding solution for the SFC request r with the randomized rounding method. The lth embedding path in Γ_r is chosen with probability q^l_r. Specifically, in line 7, a random value θ is uniformly generated in $[0, 1]$, and in line 9, if the inequality $\sum^{l}_{m=1} q^m_r \leq \theta < \sum^{l+1}_{m=1} q^m_r$ holds, the lth embedding path is selected.

It may happen that selected embedding paths are infeasible embedding solutions for SFC requests. We have designed a heuristic method to search for a feasible solution to the original ILP problem in all embedding paths. First, in line 13, the selected embedding paths P^* are sorted in descending order according to the ratio of the embedding revenue to total required bandwidth, then the embedding paths are checked for feasibility one by one. This strategy prefers SFC requests that have greater revenue-to-bandwidth requirement ratios and aims to increase the total revenue of embedding SFC requests with limited network resources. From line 16 to line 19, if an embedding path is an infeasible embedding solution for an SFC request r because of insufficient network resources in the physical network, the rest of the embedding paths in Γ_r except the selected one in P^* are tested for a feasible embedding solution, where the embedding paths are checked one by one in descending order of the path cost. We note that there will be situations in which no feasible solution exists.

Algorithm 3: Randomized Rounding based SFC Embedding

Input: Hybrid network $G^s(N^s, L^s)$, and set of SFC requests R
Output: Embedding solution P^*
//Solve the linearly relaxed LP Problem
1 Construct a linearly relaxed LP program from the original ILP problem
2 Solve the LP problem and obtain the optimal solution Opt_{LP}
//Derive an integer solution with randomized rounding method
3 **for** *each* $r \in R$ **do**
4 Call **Algorithm 1** to build an auxiliary graph G_r^{Aux} for SFC request r
5 Call **Algorithm 2** to obtain a set of embedding solutions Γ_r for SFC request r
6 Ascending sort embedding paths in the set Γ_r by path cost
7 Generate a random value θ uniformly in $[0, 1]$
8 **if** $\theta \leq \sum_{l \in \Gamma_r} q_r^l$ **then**
9 $\hat{l} = \arg\{l | \sum_{m=1}^{l} q_r^m \leq \theta < \sum_{m=1}^{l+1} q_r^m\}$
10 $P^* = P^* \cup$ the $\hat{l}$th embedding path
11 **else**
//block this request
12 The SFC request r is blocked
//Obtain feasible solutions
13 Ascending sort embedding paths in the set P^* by the ratio of the embedding revenue to the total required bandwidth
14 **for** *each embedding path* p *in the set* P^* **do**
15 **if** *the embedding path* p *is infeasible and cannot embed the corresponding SFC request* r **then**
16 **repeat**
17 Choose the embedding path p' that has the highest path cost in set Γ_r (exclude the selected path in P^*)
18 **if** *the embedding path* p' *is feasible* **then**
19 Embed the according SFC request r with p', and update network resources
until *find a feasible embedding path or there is no remaining embedding path in* Γ_r;
20 **else**
21 Embed the according SFC request r with p, and update network resources

In Algorithm 3, the complexity of solving the LP problem is determined by the number of variables and is known to be $O((|L_r||L^s||R|F)^{3.5})$ [30]. Additionally, for each SFC request, Algorithm 1 and Algorithm 2 are invoked. Algorithm 1 has a complexity of $O(|L_r||L^s|F)$, while Algorithm 2 has a complexity of $O(|F_r||L^s||\Gamma_r|F)$ due to multiple depth-first searches. Here, Γ_r represents the number of SFC embedding paths derived in the auxiliary graph, and its value is bounded by the reciprocal of the path cost granularity, which can be approximated by a constant value. Between line 13 and line 21 of Algorithm 3, the complexity of obtaining feasible solutions for SFC requests is $O(|R||\Gamma_r|)$. Ultimately, the overall complexity of the randomized rounding based SFC Embedding algorithm is $O((|L_r||L^s||R|F)^{3.5})$, primarily determined by the LP solving process.

D. Performance Analysis

The algorithm based on the randomized rounding method derives an approximate solution to the original ILP problem. In order to analyze the performance of the solution obtained by the algorithm, the following two Lemmas are used.

Lemma 1 (Chernoff Bound): Given n independent variables: $x_1, x_2, \cdots, x_n$, where $\forall x_i \in [0, 1]$. Let the expectation of the sum of these random variables: $\mu = E\left[\sum_{i=1}^{n} x_i\right]$. β is an arbitrary positive value, then we have

$$P\left[\sum_{i=1}^{n} x_i \geq (1+\beta)\mu\right] \leq exp\left(-\frac{\beta^2 \mu}{2+\beta}\right). \tag{13}$$

Lemma 2 (Union Bound): Given a finite set of n events: $A_1, A_2, \cdots, A_n$, where each event A_i happens with probability $P(A_i)$, then

$$P(A_1 \cup A_2 \cup \cdots \cup A_n) \leq \sum_{i=1}^{n} P(A_i). \tag{14}$$

To analyze the performance of the proposed SFC embedding algorithm, we introduce a discrete random variable Y_r, which represents the revenue achieved by embedding SFC request r. The value and probability distribution of Y_r are as follows.

$$Y_r = \begin{cases} p_r, & \text{with probability of } \sum_{l \in \Gamma_r} q_r^l \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Given all these random variables for SFC requests, we can formulate the total revenue achieved by embedding all SFC requests R as the random variable B, where

$$B = \sum_{r \in R} Y_r. \tag{16}$$

The expectation of the random variable B is

$$E(B) = \sum_{r \in R} E(Y_r) = \sum_{r \in R} \sum_{l \in \Gamma_r} q_r^l p_r. \tag{17}$$

On the other hand, the optimal solution to the LP problem can be represented as the weighted sum of the revenues of SFC requests. Equation (18) holds because conservation of flow is obeyed strictly when generating the embedding path.

$$Opt_{LP} = \sum_{r \in R} x_r p_r = \sum_{r \in R} \sum_{l \in \Gamma_r} q_r^l p_r \tag{18}$$

Given Equation (17) and Equation (18), the relationship between the optimal solution of the LP problem and the expectation of B is

$$E(B) = \sum_{r \in R} \sum_{l \in \Gamma_r} q_r^l p_r = Opt_{LP}. \tag{19}$$

In order to satisfy the conditions of *Lemma 1 (Chernoff-Bound)*, we need to normalize the random variables as follows:

$$\begin{cases} \frac{Y_r}{p_r^{\max}} \in (0,1] \\ B' = \sum_{r \in R} \frac{Y_r}{p_r^{\max}} \end{cases} \tag{20}$$

where $p_r^{\max}$ represents the maximum of p_r in all request R. Then we can apply *Lemma 1*, to yield

$$P\left\{B' \geq (1-\beta)E(B')\right\} \leq \exp\left(-\frac{\beta^2 E(B')}{2+\beta}\right), \tag{21}$$

where $\beta \in (0,1]$, and the relationship between B' and B is $B' p_r^{\max} = B$. So Equation(21) can be transformed into

$$P\left\{B \geq (1-\beta)E(B)\right\} \leq \exp\left(-\frac{\beta^2 E(B')}{2+\beta}\right). \tag{22}$$

At the same time, the optimal solution of the LP problem could choose the SFC request with the largest revenue for embedding, i.e., $Opt_{LP} \geq p_r^{\max}$. In which case, we have the following equation.

$$E(B') = \frac{E(B)}{p_r^{\max}} = \frac{Opt_{LP}}{p_r^{\max}} \geq 1 \tag{23}$$

So Equation (22) can be relaxed to remove the B', resulting in

$$P\{B \geq (1-\beta)E(B)\} \leq \exp(-\frac{\beta^2}{2+\beta}). \tag{24}$$

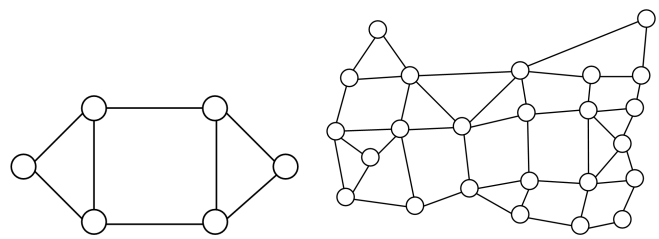
The LP problem is converted from the original ILP problem by relaxation, and the optimal result of the LP problem is an upper bound for the optimal value of the original ILP problem, i.e., $E(B) = Opt_{LP} \geq Opt_{ILP}$. Therefore, Equation (24) can be further relaxed to

$$P\{B \geq (1-\beta)Opt_{ILP}\} \leq \exp(-\frac{\beta^2}{2+\beta}). \tag{25}$$

Theorem 1: The probability that the revenue B obtained by our proposed algorithm is no less than $(1-\beta)$ times the optimal solution Opt_{ILP} of the ILP problem, is no larger than $\exp(-\frac{\beta^2}{2+\beta})$.

V. Numerical Results

In this section, we present experiments that we conducted using both small and large network topologies, as illustrated in Fig. 3. The small network topology, shown in Fig. 3(a), comprises 6 nodes and 8 links. On the other hand, for the large network topology, depicted in Fig. 3(b), we used the USNet network topology, which consists of 24 nodes and 43 links. The links here represent bidirectional connections with electronic or optical links, in which the optical network is a vertical composition of F optical layers with different wavelengths. Each electronic link is assumed to have 50 units of bandwidth resources, and a specific wavelength of an optical link has 100 units of bandwidth resources, so that an optical link has $100F$ units of bandwidth resources in total. Each network node is assumed to have 100 units of computation resources and can provide types of VNFs that consume different amounts of computation resources. The resource parameter settings for the substrate networks in this study are similar to the settings used in [8], and we assume that the capacity of a wavelength in a fiber link is twice that of an electronic link. It is worth mentioning that these parameter settings can be adjusted to other reasonable values as needed. The parameter settings provided in this paper serve as a baseline, and can be adapted to suit specific scenarios. We have conducted additional experiments with different networks and alternative resource settings, yielding similar conclusions. However, for the brevity, the simulation results from these additional experiments are not included in the paper. The CPLEX [31] commercial solver has been used to solve the ILP and LP problems in this paper.



(a) A six-node network (b) USNet network

Fig. 3: Network topologies

To generate SFC requests, we assume that the number of VNFs contained in an SFC request is a discrete random variable uniformly distributed on the set $\{2,3,4\}$, the computation resource required by a VNF node of the SFC is uniformly distributed in the real segment $[5,15]$, and the bandwidth resource required by a virtual link is also uniformly distributed in $[2,10]$.

As previously described, the sorting of SFC embedding requests in line 13 of Algorithm 3 determines the order in which the requests are embedded. Requests at the front are more likely to be successfully embedded because of the availability of sufficient network resources. This heuristic sorting approach has a significant impact on the performance of the proposed algorithm. However, there are alternative sorting methods that may produce different effects on performance. Accordingly, we modify our sorting approach and compare the resulting algorithms to assess their relative performance. Two new heuristic algorithms that are slight modifications of the proposed algorithm are discussed. Specifically, in line 13 of Algorithm 3, the embedding paths are sorted in descending order of the ratio of embedding revenue and total required bandwidth; this we name the PB-RRSE (profit over bandwidth-randomized rounding select embedding) algorithm. We only modify the sorting operation in line 13, which is to sort the embedding paths in descending order of embedding revenues. We call this modification the P-RRSE (profit-randomized rounding select embedding) algorithm. In a similar vein, we

propose a modification in which the sorting is changed to be ascending sorting of total required bandwidth; this modification is called the B-RRSE (bandwidth-randomized rounding select embedding) algorithm. In addition, the optimal solution of the original ILP formulation will be used to benchmark the performance for the approximate algorithms. To ensure that the results of our comparison are adequate, we have taken several measures. Firstly, the inclusion of the ILP method as a benchmark allows for a performance comparison between the proposed algorithms and the optimal solution. Secondly, the three proposed SFC embedding algorithms exhibit different but reasonable performances, as validated through extensive simulations with diverse network configurations. Finally, the analytical performance bounds provided for these algorithms offer performance guarantees, further enhancing their credibility.

In our optimization problem, the primary objective is to maximize revenue that directly reflects the financial gain obtained from accommodating SFC requests. However, we also recognize the importance of considering a more equitable performance metric, namely the acceptance ratio, which is the proportion of SFC requests that are successfully embedded within the available network resources. The acceptance ratio provides a measure of quality of service and fairness; particularly, when all requests are equal and revenue is not a differentiating factor. In this paper, we have considered both revenue maximization and the (overall) acceptance ratio to assess the consequences of our revenue-focused policy on this more equitable performance measure. By considering the acceptance ratio alongside revenue, we aim to gain insights into the impact of our revenue maximization strategy on the overall acceptance ratio within the system. Note that, in this paper, the (overall) acceptance ratio is not the same as the acceptance ratio observed by the user which depends on the user's payment. Generally, as users pay more, they tend to observe a lower acceptance ratio. In this paper, the acceptance ratio has positive correlation with the revenue. Although the revenue and the acceptance ratio are related, they represent different aspects of performance. While a high acceptance ratio is desirable, it does not necessarily guarantee a high revenue if the accepted SFC requests have low associated revenues. We may take actions that will lower the acceptance ratio but will increase revenue if we choose, for example, to reject many low-income requests and instead accept a few high-income requests. In our evaluation, we consider both revenue and acceptance ratio as separate performance metrics to provide a comprehensive analysis of the proposed algorithms. By examining these metrics, we can gain insights into the effectiveness of the algorithms in terms of both revenue and overall acceptance ratio.

In Fig. 4, the revenue of embedding SFC requests is compared, where the revenue value is the average value of ten experiments for each scenario. The ILP method has the highest value, followed by PB-RRSE, P-RRSE and B-RRSE. The revenue of the P-RRSE algorithm is higher than that of the B-RRSE algorithm. This is because the B-RRSE algorithm only considers the bandwidth requirement in sorting, and so may prefer to embed the SFC requests with small bandwidth

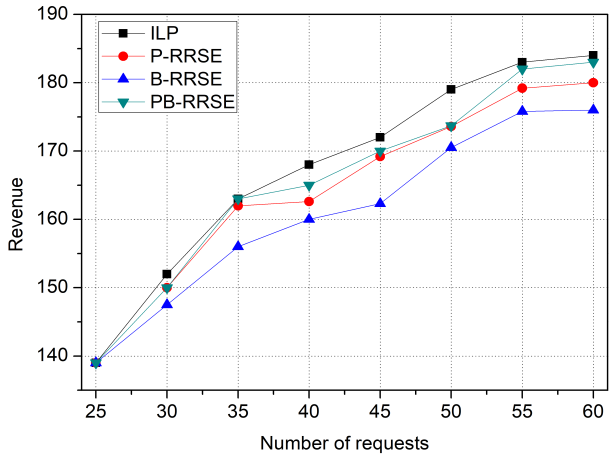


Fig. 4: Revenue in the six-node network

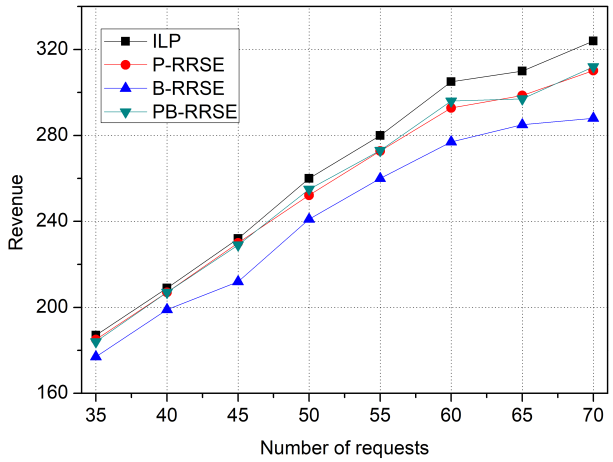


Fig. 5: Revenue in USNet network

requirements, resulting in the failure of some high-revenue requests with large bandwidth requirements and finally leading to lower revenue for the whole system. We also observe that the revenue of the P-RRSE algorithm is close to that of the PB-RRSE algorithm. This is because, in some cases, preferring the SFC requests with high revenues may lead to their embedding but resulting in the consumption of vast bandwidth resources. Compared with the P-RRSE and B-RRSE algorithms, the PB-RRSE algorithm considers the embedding revenue and the bandwidth requirements simultaneously in sorting. This may give preference to embedding the SFC requests with high embedding revenue and small bandwidth requirements, leading to high revenue and low resource consumption. Meanwhile, the revenue of the PB-RRSE algorithm and that of the ILP method are pretty close, which verifies the performance of the PB-RRSE algorithm.

Similar observations can be found in the USNet network in Fig. 5, where the ILP method has the highest value, followed by PB-RRSE, P-RRSE and B-RRSE. We also note that using ILP for solving the USNet network can only be performed with a long running time. This makes ILP prohibitive for larger problems.

The acceptance ratio is the ratio of the number of accepted

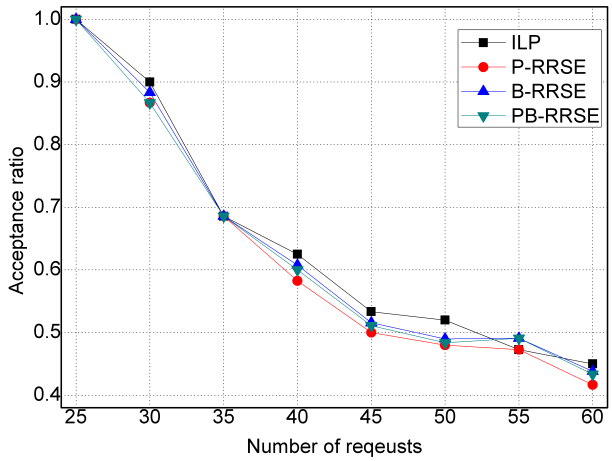


Fig. 6: Acceptance ratio in the six-node network

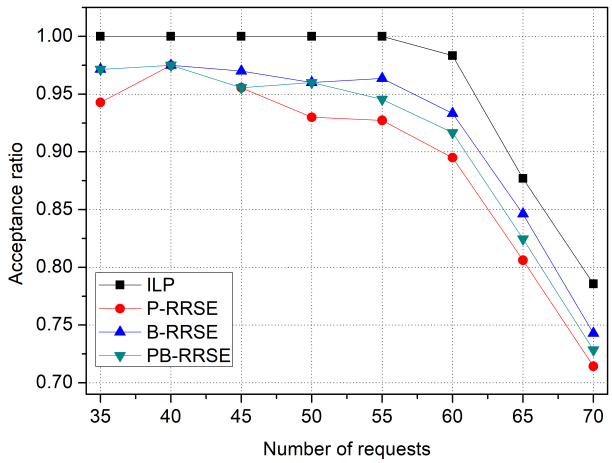


Fig. 7: Acceptance ratio in USNet network

requests to the total number of requests, which is important for evaluating the performance of algorithms. Figs. 6 and 7 show the acceptance ratios of the three heuristic algorithms and the ILP method in the six-node network and USNet network, respectively. The acceptance ratio is often considered an important performance measure because of its significance as a service quality level provided by a network for customers. Although in this paper, we aim to optimize the total revenue, we are also interested in measuring the acceptance ratio, because its QoS evaluation should not be significantly adversely affected by our revenue-based objective function.

In Fig. 6, the acceptance ratio of the ILP method is the highest among the four, except that when the number of requests is 55, it is lower than PB-RRSE and B-RRSE algorithms. This high acceptance ratio performance of the ILP method is associated with the high computation complexity of the ILP solver, which can search the entire variable region for embedding solutions. When the number of requests is 55, the ILP method has a lower acceptance ratio than the PB-RRSE; this is because the ILP method tries to improve the objective value (revenue) and selects a high-revenue request to embed (high-revenue request usually requires more resources), but more SFC requests have to be blocked because of lack of resources. On the other hand, the PB-RRSE selects more low-revenue requests to embed instead of one high-revenue request. Fig. 7 shows the acceptance ratio comparison for the USNet network. The observations are almost the same as that in Fig. 6, except that Fig. 7 has a highly stable acceptance ratio when there are a small number of requests. This is because of the abundance of the network resource relative to the small number of requests.

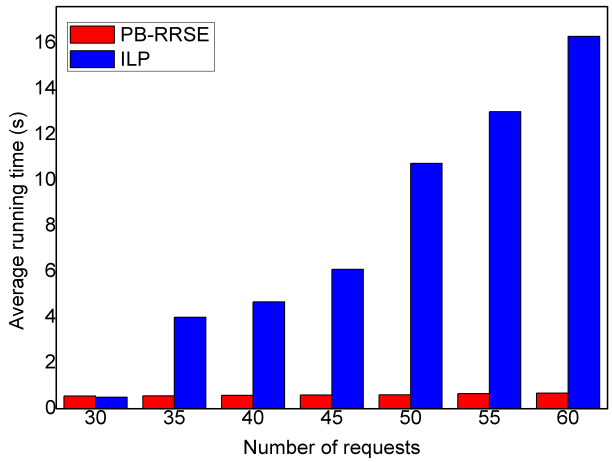


Fig. 8: Running time in the six-node network

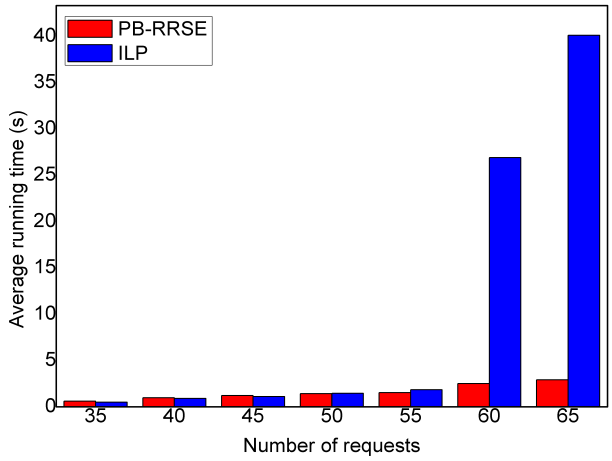


Fig. 9: Running time in USNet network

After investigating the revenue and acceptance ratio of the four methods, we find that the performance of the PB-RRSE algorithm is very close to the ILP method, which provides the optimal solution to the original problem. Next, we do a comparison of the solving time between the PB-RRSE algorithm and the ILP method to show the time efficiency of the PB-RRSE algorithm.

Figs. 8 and 9 show a comparison of the average running time of the ILP method and the PB-RRSE algorithm in the six-node network and USNet network, respectively. In the two figures, as the number of SFC requests increases, the average running time of the PB-RRSE algorithm remains low and stable. The running time, of the ILP solver, increases sharply. The reason is that the PB-RRSE algorithm contains the processes of linear relaxation, auxiliary graph construction,

embedding path generation, randomized rounding, requests sorting, and rerouting after embedding failure, where each process is polynomial, resulting in the entire algorithm being polynomial. The ILP solver tries to solve the NP-hard problem, which results in a significant increase in the running time as the problem size increases. It is noted that in Fig. 9, the experiments with 70 and more SFC requests are also carried out, where the running time of the experiment with 70 requests takes about 30 minutes and the one with 75 or more requests can not be solved out within hours, whereas the PB-RRSE algorithm derives the solution within 3 seconds. For the sake of brevity, these values are not shown in the figure.

In the context of SFC embedding, the allocation of both bandwidth and computation resources forms the foundation of the embedding process. Computation resources are deployed at network nodes to provide VNFs for SFC requests. Typically, network resources are limited, and it is only through the implementation of efficient algorithms that a high network revenue can be achieved. The quantity of computation resources available at network nodes serves as a constraint in the SFC embedding problem, directly impacting the performances of algorithms. Simulations are conducted to verify the effectiveness of the algorithms by considering different computation resource scenarios in the network. Table II presents four scenarios, each representing a specific amount of computation resources (100 units, 150 units, 200 units, and 250 units). In each scenario, each network node is allocated the specific amount of computation resources, and there are 70 SFC requests for each experiment. For the scenarios we considered, according to Table II, it is observed that increasing the computation resources leads to an increase in revenue. This can be attributed to the fact that a greater availability of computation resources allows for accommodating a higher number of SFC requests. However, it should be noted that as computation resources become abundant, other constraints, such as bandwidth resources, start to emerge as the limiting factor impacting system performance. Consequently, despite the surplus of computation resources, the revenue plateaus due to the constraints imposed by bandwidth resources.

TABLE II: Revenues vs. Computation resources in each node

Computation resources (units)	100	150	200	250
ILP	322.2	333.6	358.5	358.5
P-RRSE	316.5	321.1	355.4	356.3
B-RRSE	295.4	314	348.4	356
PB-RRSE	318.4	319.7	352.6	358.2

In Table III, we present simulations conducted with varying bandwidth resources in each electronic link, similar to the approach taken in Table II. Specifically, we set bandwidth resource values of 50 units, 100 units, 150 units, and 200 units for the four different scenarios. In Table III, we observe a consistent pattern: an increase in the availability of bandwidth resources corresponds to an increase in revenue, mirroring the relationship seen with computation resources in Table II. This revenue increase can be attributed to the fact that a greater availability of bandwidth resources enables the fulfillment of a higher number of SFC requests, leading to increased revenue. However, as bandwidth resources become abundant, computation resources begin to emerge as the limiting factor. Overall, the findings in Table III align with those of Table II, underscoring the importance of both bandwidth and computation resources in the context of SFC provisioning.

TABLE III: Revenues vs. bandwidth resources in each electronic link

Bandwidth resources (units)	50	100	150	200
ILP	322.2	333.4	335.2	338.6
P-RRSE	316.5	320.2	321.1	325.7
B-RRSE	295.4	310.8	311.4	315.2
PB-RRSE	318.4	322.9	323.6	325.2

To further evaluate the effectiveness of the proposed heuristic algorithms, we conducted experiments on a larger randomly generated network comprising 40 nodes and 100 links. In this network, each node is connected to at least three neighboring nodes, ensuring a minimum degree of three for each node. The number of VNFs in an SFC request is uniformly distributed from the set $\{3, 4, 5, 6, 7\}$. It is important to note that we did not specify the particular VNFs or service functions (SFs) to form the SFC requests. This is intentional as the approach can be easily extended to include any other VNF instances. The reason behind this flexibility is that VNFs are software-based entities that can be instantiated without distinction and scaled up or down using NFV technology. We assume that the computation costs of the SFs are proportional to their computation requirements, which, in turn, are proportional to the revenues for network operators. This assumption aligns with practical considerations. In this paper, we focus on a network with limited computation and network resources. This justifies our objective of accommodating more SFC requests to maximize total revenue, as it also reflects the cost associated with embedding SFCs. Given the significantly long solving time, we found the ILP method to be impractical for this particular network. Consequently, our focus shifted to evaluating the performance of the proposed heuristic algorithms. Fig. 10 illustrates the network revenues obtained for different SFC requests, exhibiting a similar trend to that observed in previous networks. For this network example, the PB-RRSE method achieved the highest revenue values, followed by P-RRSE and B-RRSE. Furthermore, Fig. 11 presents the acceptance ratios for different numbers of SFC requests, showing a similar trend to the previous networks. For our example, the PB-RRSE method achieved acceptance ratio values falling between those of P-RRSE and B-RRSE. In addition, all the algorithms provided a solution in less than three seconds.

VI. Conclusion

This paper investigates the SFC embedding problem in hybrid optical-electronic networks with a given set of SFC requests. We provide an ILP formulation and design a heuristic algorithm based on the randomized rounding method for scalability, as the original problem is NP-hard. Our algorithm involves linear relaxation, building auxiliary graphs, generating embedding paths, randomized rounding, and re-selection of embedding paths to construct a feasible integer approximate

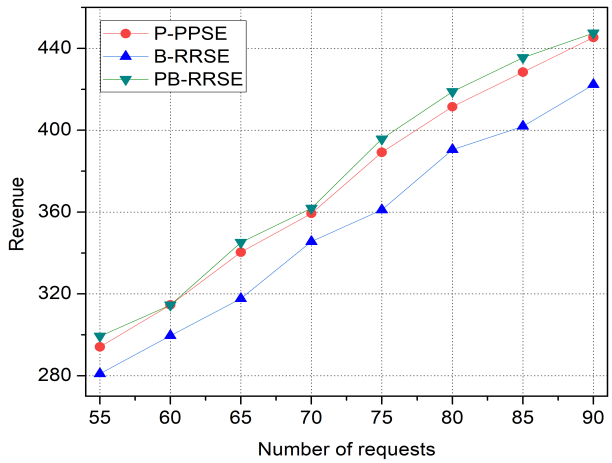


Fig. 10: Revenue vs. number of requests in a 40-node randomly generated network

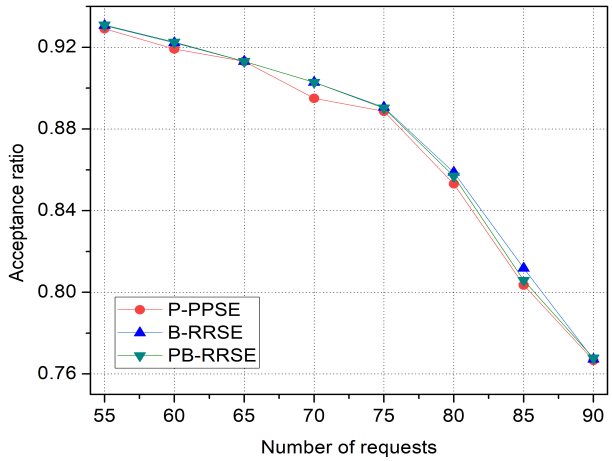


Fig. 11: Acceptance ratio vs. number of requests in a 40-node randomly generated network

solution for the original embedding problem. Furthermore, we provide a performance analysis of the algorithm and verify its performance through simulations.

References

[1] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "NFV: State of the art, challenges and implementation in next generation mobile networks (vEPC)," *IEEE Network*, vol. 28, no. 6, pp. 18–26, 2014.

[2] M. Chiosi, D. Clarke, P. Willis, and A. Reid, "Network functions virtualisation introductory white paper," 2012.

[3] S. Kumar, M. Tufail, S. Majee, C. Captari, and S. Homma, "Service Function Chaining Use Cases In Data Centers," Internet Engineering Task Force, Internet-Draft draft-ietf-sfc-dc-use-cases-06, Feb 2017.

[4] N. M. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks*, vol. 54, no. 5, pp. 862–876, 2010.

[5] A. Fischer, J. F. Botero, M. T. Beck, H. de Meer, and X. Hesselbach, "Virtual network embedding: A survey," *IEEE Commun. Surveys Tuts*, vol. 15, no. 4, pp. 1888–1906, 2013.

[6] B. Yi, X. Wang, K. Li, S. k. Das, and M. Huang, "A comprehensive survey of network function virtualization," *Computer Networks*, vol. 133, pp. 212–262, 2018.

[7] D. Bhamare, R. Jain, M. Samaka, and A. Erbad, "A survey on service function chaining," *J. Netw. Comput. Appl.*, vol. 75, pp. 138–155, 2016.

[8] D. Zheng, C. Peng, X. Liao, and X. Cao, "Toward optimal hybrid service function chain embedding in multiaccess edge computing," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6035–6045, 2020.

[9] J. G. Herrera and J. F. Botero, "Resource allocation in NFV: A comprehensive survey," *IEEE Trans. Netw. Service Manag*, vol. 13, no. 3, pp. 518–532, Sep. 2016.

[10] H. A. Alameddine, S. Sebbah, and C. Assi, "On the interplay between network function mapping and scheduling in VNF-based networks: A column generation approach," *IEEE Trans. Netw. Service Manag*, vol. 14, no. 4, pp. 860–874, 2017.

[11] J. Fan, Z. Ye, C. Guan, X. Gao, K. Ren, and C. Qiao, "GREP: Guaranteeing reliability with enhanced protection in NFV," in *Proceedings of the 2015 ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization*, Aug 2015, pp. 13–18.

[12] N. Bouten, J. Famaey, R. Mijumbi, B. Naudts, J. Serrat, S. Latré, and F. De Turck, "Towards NFV-based multimedia delivery," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, May 2015, pp. 738–741.

[13] M. F. Bari, S. R. Chowdhury, R. Ahmed, and R. Boutaba, "On orchestrating virtual network functions," in *2015 11th International Conference on Network and Service Management (CNSM)*, Nov 2015, pp. 50–56.

[14] D. Bhamare, M. Samaka, A. Erbad, R. Jain, L. Gupta, and H. A. Chan, "Optimal virtual network function placement in multi-cloud service function chaining architecture," *Computer Communications*, vol. 102, no. C, pp. 1–16, 2017.

[15] Y. Jia, C. Wu, Z. Li, F. Le, and A. Liu, "Online scaling of NFV service chains across geo-distributed datacenters," *IEEE ACM. Trans. Netw*, vol. 26, no. 2, pp. 699–710, 2018.

[16] D. Zheng, H. Gu, W. Wei, C. Peng, and X. Cao, "Network service chaining and embedding with provable bounds," *IEEE Internet Things J.*, vol. 8, no. 9, pp. 7140–7151, 2021.

[17] D. Zheng, G. Shen, X. Cao, and B. Mukherjee, "Towards optimal parallelism-aware service chaining and embedding," *IEEE Trans. Netw. Service Manag.*, vol. 19, no. 3, pp. 2063–2077, 2022.

[18] M. Laroui, H. I. Khedher, H. Moungla, and H. Afifi, "Service function chains multi-resource orchestration in virtual mobile edge computing," *Computer Networks*, vol. 224, p. 109582, 2023.

[19] Y. Liu and J. Zhang, "Service function chain embedding meets machine learning: Deep reinforcement learning approach," *IEEE Trans. Netw. Service Manag.*, pp. 1–1, 2024.

[20] D. Zheng, G. Shen, Y. Li, X. Cao, and B. Mukherjee, "Service function chaining and embedding with heterogeneous faults tolerance in edge networks," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 3, pp. 2157–2171, 2023.

[21] T. Lin, Z. Zhou, M. Tornatore, and B. Mukherjee, "Demand-aware network function placement," *J. Lightw. Technol.*, vol. 34, no. 11, pp. 2590–2600, 2016.

[22] M. Xia, M. Shirazipour, Y. Zhang, H. Green, and A. Takacs, "Optical service chaining for network function virtualization," *IEEE Commun. Mag*, vol. 53, no. 4, pp. 152–158, 2015.

[23] W. Fang, M. Zeng, X. Liu, L. Wei, and Z. Zhu, "Joint spectrum and IT resource allocation for efficient VNF service chaining in inter-datacenter elastic optical networks," *IEEE Commun. Lett*, vol. 20, no. 8, pp. 1539–1542, 2016.

[24] M. Zeng, W. Fang, and Z. Zhu, "Orchestrating tree-type VNF forwarding graphs in inter-DC elastic optical networks," *J. Lightw. Technol.*, vol. 34, no. 14, pp. 3330–3341, 2016.

[25] S. Zhang and K. L. Yeung, "Efficient embedding of service function chains in space-division multiplexing elastic optical networks," *Computer Networks*, vol. 233, p. 109869, 2023.

[26] J. Yu, W. Zheng, W. Shao, H. Chen, D. Zheng, B. Chen, and J. Wu, "Cost-effective service function chain mapping approaches in edge-cloud elastic optical networks," in *Proc. Asia Communications and Photonics Conference*, 2022, pp. 1160–1162.

[27] Y. Wang, L. Nguyen, and Q. Hu, "Network function virtualization in elastic optical networks," *J. Lightw. Technol.*, vol. 41, no. 16, pp. 5183–5192, 2023.

[28] R. Lin, B. Ding, S. Luo, and M. Zukerman, "Service function chaining embedding in hybrid optical-electronic networks," *J. Lightw. Technol.*, vol. 40, no. 15, pp. 4922–4933, 2022.

[29] P. Raghavan and C. D. Tompson, "Randomized rounding: A technique for provably good algorithms and algorithmic proofs," *Combinatorica*, vol. 7, no. 4, pp. 365–374, 1987.

[30] N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica*, vol. 4, pp. 373–395, 1984.

[31] IBM ILOG CPLEX, IBM, Inc., Mountain View, CA[Online], Available: https://www.ibm.com/products/ilog-cplex-optimization-studio.