IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 4, 2019, accepted December 13, 2019, date of publication December 17, 2019, date of current version December 31, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2960395

Bi-Level Programming Model and Algorithm for VNF Deployment With Data Centers Placement

HEJUN XUAN[1], SHIWEI WEI[2], YAN FENG[1,3], DAOHUA LIU[1,3], AND YANLING LI[1,3]
[1]School of Computer and Information Technology, Xinyang Normal University, Xinyang 464000, China
[2]School of Computer and Technology, Guilin University of Aerospace Technology, Guilin 541000, China
[3]Key Laboratory of Analysis and Application of Education Big Data, Xinyang Normal University, Xinyang 464000, China

Corresponding author: Hejun Xuan (xuanhejun0896@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 31900710, Grant 61572391, and Grant 61572417, in part by the Science and Technology Department of Henan Province under Grant 182102210132 and Grant 182102210537, in part by the Innovation Team Support Plan of University Science and Technology of Henan Province under Grant 19IRTSTHN014, in part by the Guangxi Natural Science Foundation of China under Grant 2016GXNSFAA380226, in part by the Guangxi Young and Middle-aged Teachers' Basic Ability Improvement Foundation of China under Grant 2017KY0866, in part by the Internet of Things and Big Data Application Research Foundation of Guilin University of Aerospace Technology under Grant KJPT201809, and in part by the Nanhu Scholars Program for Young Scholars of XYNU.

ABSTRACT Virtual network function (VNF) can provide various network services and is widely deployed in inter-data centers elastic optical networks (Inter-DC EONs). Routing and VNF deployment for VNF service chain (VNF-SC) in Inter-DC EONs is a very important and well-known NP-hard problem. For this problem, if determining the number and locations of data centers is additionally considered, it will be more complicated. In this paper, we investigate a network planning problem in Inter-DC EONs by determining all these factors, i.e, by determining not only the optimal routing and the optimal VNF deployment for VNF-SCs, but also the optimal number and locations of data centers. To achieve this purpose, we first establish a bi-level programming model in which the leader's objective is to minimize the number of data centers and find the best locations of data centers so that we can get a balanced VNF deployment on data centers. To determine the optimal routing and VNF deployment for VNF-SCs, the follower's objective is to minimize the maximum index of used frequency slots and the number of used frequency slots. Then, to solve the proposed model effectively, tailor-made crossover, mutation and local search operators are designed, and based on these operators, an efficient bi-level hybrid memetic algorithm (BiHMA) is proposed. Finally, to test the effectiveness of the proposed model and the efficiency of the proposed algorithm, the simulation experiments are conducted on two widely used networks, and experimental results indicate that the proposed algorithm has a higher efficiency than compared algorithms.

INDEX TERMS EONs, data centers placement, bi-level optimization, Memetic algorithm.

I. INTRODUCTION

In recent years, service providers are facing the problem to deploy some new network services flexibly and effectively. To realize this, researchers developed the network function virtualization (NFV) to satisfy the demands of users [5], [31], [44]. NFV utilizes virtualization technology to decouple network functions from dedicated hardware into virtual network functions so that these functions can run as software images on commodity hardware as well as custom-built hardware [6], [11], [13]. With NFV technology, traditional hardware-based network appliances are replaced by software-based virtual network functions (VNFs), and VNFs can be realized on the datacenters (i.e., VNF service chain, VNF-SC) [10], [46]. The deployment of new network services can be easily realized by routing data traffic through a series of VNFs on datacenters [14], [41]. In general, VNF-SC has many advantages, such as high bandwidth capacity and low power consumption [32], [47], [49]. So, VNF-SC is especially beneficial for inter-DC networks. However, it is difficult and challenging to use the VNF-SCs.

The associate editor coordinating the review of this manuscript and approving it for publication was Emanuele Crisostomi.

In recent years, some researches have been done on network planning problems using the VNF-SCs, which mainly focused on the service chain routing and VNF deployment problem (e.g., [3], [14], [37], [38]). To minimize the sum of cloud resource, bandwidth and deployment costs, an integer linear problem is formulated, and an efficient heuristic approach allowing for a remarkable computational complexity reduction is designed [8]. To minimize the cost of the VNF deployment in an inter-DC EONs, an integer linear programming model was first established, and then three heuristic algorithms were proposed to solve the integer linear programming model effectively [47]. To minimize the total cost of the energy consumption and the revenue loss due to QoS degradation, an efficient algorithm, which is based on the back-to-back strategy, was designed [37]. To solve service chain and resource allocation problem, a mixed-integer linear programming model was proposed [38], and a heuristic-based algorithm, which consists of two sub-algorithms: one-hop optimal traffic scheduling algorithm and VNF chain composition algorithm, was proposed. Literature [9] minimizes the sum of three cost components include the cloud resource cost, the bandwidth cost, and the reconfiguration cost, reconfiguration costs are characterized by the revenue loss of a network operator due to the bit loss. Li and Qian [27] formulated the VNF deployment problem as an integer linear programming model and proposed a simulated annealing based heuristic to get approximate solutions in shorter time. Considering elastic optical networking and DC capacity constraints, an effective algorithm based noncooperative mixed-strategy gaming approach is proposed [7]. For the sake of addressing the relatively long setup latency and complicated network control, a provisioning framework with resource pre-deployment to resolve the aforementioned challenge is designed [25]. The proposed gaming model enables tenants to compete for VNF-SC provisioning services due to revenue and quality-of-service incentives and therefore can motivate more reasonable selections of provisioning schemes. Marouen *et al.* [29] proposed an algorithm based on eigenvalue decomposition for the VNF deployment in EONs for the sake of satisfying user's requirements and maximizing provider's revenue. To provide an NFV network model suitable for ISP operations, authors in [1] defined the generic VNF chain routing optimization problem and devise a mixed integer linear programming formulation. To improve the overall allocation performance of deploying service chains with server affinity, collocation, and latency constraints, a strategy based on graph partitioning and game theory was proposed [23]. Ahvar *et al.* [2] formulated the problem of VNF placement as an integer linear programming model and proposed a cost-efficient centrality-based VNF placement algorithm to minimize the provider cost by determining the optimal number of VNFs and their locations. In two domain EONs, literature [40] proposed an integer linear programming model, and designed two time-efficient heuristic algorithms to minimize the total resource cost of VNF provisioning. To investigate the importance and the relation among the link, the server usage in VNF placement and the path selection, a systematic method was proposed which can elastically tune the proper link and server usage of each demand based on network conditions and demand properties [22]. To minimize the spectrum and switch port usage, an integer linear programming model was established to solve the resource allocation problem in intra-DC EONs [33].

Motivation: Existing studies studied the routing and VNF deployment for VNF-SC, and DC placement problem dependently. In this paper, we investigate a network planning problem in EONs by considering all these factors, i.e.,we should determine not only the optimal routing and VNF deployment scheme for VNF-SC, but also the optimal number and location of datacenters. To solve the problem, we should determine the number and location ode the DC, i.e., DC placement problem. Then, routing and VNF deployment scheme for each VNF-SC should be determined. In general, the number and location of DC has a great effect on the efficiency of routing and VNF deployment scheme for VNF-SC. Meanwhile, routing and VNF deployment scheme will affect the load of DC. If the routing and VNF deployment scheme is optimal, all the VNF can deployed on DCs balanced. However, the number and location of DC have an effect on the balance of load in all DCs. So, routing and VNF deployment scheme can affect the objective of DC placement problem. DC placement problem and routing and VNF deployment problem has a priority order, and the two problem can effect interact on each other. So, DC placement problem and routing and VNF deployment problem has a hierarchical property. That is to say, it is necessary to consider the relationship between routing and VNF deployment problem for VNF-SC and DC placement problem. That is to say, we should investigate routing and VNF deployment problem for VNF-SC and DC placement problem jointly. It is not suitable to solve this problem with a global constrained optimization model. Considering the hierarchical property in the process of determining the optimal routing and VNF deployment as well as the DC placemen, we establish a bi-level programming model which can well reflect this hierarchical property and consider all factors. The major contributions of this study are summarized as follows:

- We give an effective scheme to determine not only the optimal routing and VNF deployment for VNF-SCs, but also the optimal number and locations of datacenters, i.e., the optimal datacenters placement.
- we establish a bi-level programming model which can well reflect this hierarchical property in the process of determining the optimal routing and VNF deployment as well as the DC placement.
- We design an effective bi-level hybrid memetic algorithm (BiHMA) with a novel encoding scheme and tailor-made crossover, mutation and local search operators to solve the proposed model.

The rest of this paper is organized as follows. Description of Bi-level optimization problem and Memetic algorithm are given in section II. Section III describes the problem formulation, and establishes the optimization model. To solve the optimization model effectively, we propose a memetic algorithm with tailor-made operators in section IV. To evaluate the algorithm proposed, simulation experiments are conducted, and the experimental results are analyzed in Section V. The paper is concluded with a summary in Section VI.

II. BI-LEVEL OPTIMIZATION PROBLEM AND MEMETIC ALGORITHM

A. BI-LEVEL OPTIMIZATION PROBLEM

To solve the context of unbalanced economic markets problem, Von Stackelberg introduced the bi-level mathematical problem (BLPP), which can be viewed as a static version of the non-cooperative and two-person-game [35]. Bi-level mathematical is a technique which can be used to modeling decentralized decision problem. It consists of the leader-level and follower-level objectives [21]. If x and y denote the leader's decision variables and follower's decision variables, generic bi-level mathematical problem can be written as

$$\min_{x} U(x, y)$$

where y is obtained by solving the follower level optimization problem

$$\min_{y} L(x, y)$$

In this mathematical model, the evaluation of the leader-level objective function $U(x, y)$ requires solving the follower-level objective function $L(x, y)$. That is to say, leader decision maker cannot minimize its objective without the reactions of the followers considered. There are numerous of algorithms focusing on solving bi-level mathematical problem, like methods based on vertex enumeration and metaheuristics [20].

B. MEMETIC ALGORITHM

Memetic algorithm, which adds the local search operator to genetic algorithm, has been proven to be an effective technique for many hard problems such as production-distribution planning problems [15], transportation and network design [45], Scientific workflow scheduling [28], [30], [48], task scheduling in cloud computing [39], [42], [43]. There are some conceptions, including encoding, decoding, individual, population, crossover operator, mutation operator and local search operator, fitness function etc. To make it more understood easily, we will introduce these conceptions detailed as follows:

Encoding: Generally speaking, a suitable encoding scheme, which encodes the solutions in problem domain to a chromosome, is much more significant. A better encoding scheme will make the search easier by limiting the search space and converge to the global optimal solution rapidly.

Decoding: Decoding scheme decodes the chromosome to a solution in problem domain.

Individual: A chromosome is an individual. That is to say, an individual can be translated to a solution of the problem.

Population: Memetic algorithm is a swarm intelligence algorithm. A population includes some individuals. In general, the number of individuals in a population is called population size.

Crossover operator: Crossover is an important operator in memetic algorithms. It can generate new offspring by combining two parents. The offspring generated are very possible to be better than both of the parents if the specific characteristic of the parents is used.

Mutation operator: Mutation is a operator which can change some gene values in the parent individual to a new state. A better mutation operator can produce entirely novel offspring individuals and improve diversity of the population. With these new individuals, the memetic algorithm may obtain a better solution than the previous one.

Local search operator: Mutation is an essential operator in memetic algorithm, and it can help the individuals to escape from the local optimum.

Fitness function: Fitness function is an indicator which can evaluate quality of the individuals.In general, a fitness function is derived from the objective function of the optimization model.

III. PROBLEM DESCRIPTION AND MATHEMATICAL MODELING

A. PROBLEM DESCRIPTION

We use an undirected graph $G(V, E)$ to denote an EONs, where $V = \{v_1, v_2, \cdots, v_{N_V}\}$ represents the set of the nodes in the network with N_V being the number of nodes and v_i is the i-th optical node, respectively. Each node is a network device such as gateway, router and switch, etc. $E = \left\{l_{ij} | v_i, v_j \in V\right\}$ represents the set of optical links with l_{ij} being the link between node v_i and node v_j, and N_E is the number of links in EONs. Each link has N_F frequency slots, and the indexs of frequency slots on each link are $1, 2, \cdots, N_F$. Now, there are a set of datacenters to be connected to EONs and some tasks to be completed in EONS.

To save the cost, our first problem is how to use as small number of data centers as possible to be connected to nodes of EONs? and determine which nodes in EONs connecting to the used data centers.

When the number of used data centers is determined, we have to select the same number of nodes in EONs to connect to these used data centers. In general, each selected node must connect to only one used datacenter (If a selected node connects to a used datacenter, it can only connect to this datacenter) and vice versa (one used datacenter must be connected to only one selected node), i.e., each selected node must uniquely connect to one used datacenter and each used datacenter also must be uniquely connected to one selected node. When a selected node connects to a used datacenter, it means that the node has all resources and functions of the datacenter without any cost in EONs. In this way, we can consider this node connecting to a used datacenter as a DC-node which has all resources and functions of the datacenter without any cost. Thus, the EONs with the DC-nodes forms a new network called Inter-DC EONs.

Each DC-node can realize some virtual network functions such as firewall, deep package inspection, network monitoring, etc., and other nodes can not realize any of these functions. The set of these virtual network functions is denoted by $VNF = \{VNF_1, VNF_2, \cdots, VNF_{N_{vnf}}\}$, where VNF_i is the i-th virtual network function and N_{vnf} is the number of total virtual network functions.

For the network Inter-DC EONs, there are some connection requests. Each connection request corresponds to a task called a virtual network function service chain denoted as a VNF-SC. A VNF-SC is such a task, in which we have to choose a path from a given source node to a given destination node to send a given data, and choose some DC-nodes in the path to realize some virtual network functions. To complete the task, we have to do the following things: 1) select a path from the source node to the destination node to send the given data (i.e., select the inter-nodes including the DC-nodes in the path. This is called path selection); 2) select the frequency slots in each link of the path to send the given data (this is called spectrum assignment); 3) assign which DC-nodes in the path to realize which VNFs (this is called VNF deployment)? Doing these three things is called one task scheduling.

Now we have a set of tasks, i.e., a set of VNF-SCs, denoted by $T = \{T_1, T_2, \cdots, T_{N_T}\}$, where N_T is the number of tasks (VNF-SCs), and T_k is the k-th task (VNF-SC). We can represent task T_k as $T_k = (s_k, d_k, VNF_k^T, b_k)$, where s_k and d_k represent that we have to choose a path to send the given data from the source node s_k to the destination node d_k by occupying the some frequency slots in each link of the path (the bigger the amount of the given data is, the more the frequency slots will be occupied to send the data), and $VNF_k^T = \{VNF_{k_1}, VNF_{k_2}, \cdots, VNF_{k_{M_k}}\}$ is the set of VNFs to be realized in task T_k with M_k being the number of VNFs in VNF_k^T, i.e., we have to choose some DC-nodes in the selected path to realize these VNFs. $b_k = (b_k^0, b_k^1, \cdots, b_k^{M_k})$ is the numbers of frequency slots of T_k required to send the given data, where b_k^0 is the number of frequency slots occupied by the initial data in T_k. When the initial data for T_k from the source node is sent to a DC-node which will realize VNF_{k_1} and after VNF_{k_1} is realized on this DC-node, the size of the data will change and the changed data will occupy b_k^1 frequency slots. The changed data will continue to be sent to the next DC-node. Generally, let b_k^m denote the number of frequency slots occupied by the changed data after VNF_{k_m} is realized. In addition, we have $\forall VNF_k^T \subseteq VNF (1 \leq k \leq N_T)$.

Thus, the second problem is that: For a set of given tasks, how to make the task scheduling for these tasks?

By combining the first and second problems, the whole problem can be seen as a two phase problem and can be summarized as follows: For the first phase, for a set of given data centers, we have to use as small number of data centers as possible and determine the DC-nodes and form the network Inter-DC EONs (data centers placement). For the second phase, for a set of given tasks, we have to determine the task scheduling scheme to schedule all given tasks such that some objectives are optimal.

Note that in the first phase, if the number of used datacenters (DC-nodes) is large, it will be costly, otherwise, some VNF-SCs will be blocked. Similarly, the placement of used datacenters (DC-nodes) can also affect the balance of VNFs deployment on the DC-nodes. So, the objective in the first phase is to determine the optimal scheme of datacenter placement (i.e., to minimize the number of DC-nodes and find the optimal locations of DC-nodes). In the second phase, the path selection (routing), VNF deployment and spectrum assignment will affect the amount of the resource used (i.e., the maximum index of used frequency slots and the number of used frequency slots). To save the resource, the objective in the second phase is to determine the optimal schemes of path selection, VNF deployment and spectrum assignment for all tasks (i.e., to minimize the maximum index of used frequency slots and the number of used frequency slots). Since the second phase must be done after the first phase, and the scheme of the first phase has an effect on the second phase. That is to say, there is a hierarchical relation between the first phase and the second phase. So, this problem can be modelled by a bi-level programming model, and the first phase and the second phase can be regarded as the leader decision and the follower decision, respectively. The datacenter placement can be regarded as the leader's variable, while the path selection, spectrum assignment and VNF deployment can be treated as the follower's variables.

B. MATHEMATICAL MODELING

In this section, we will set up a bi-level programming model for the problem.

1) LEADER'S OBJECTIVE

The smaller the number of the used DC-nodes is,the smaller the cost will be. Thus, one objective of leader is to minimize the number of used DC-nodes N_{DC}. Let

$$x = (x_1, x_2, \cdots, x_i, \cdots, x_{N_V})$$

be the scheme of datacenter placement, i.e., $x_i = 1$ if and only if node v_i is selected as DC-node (i.e., node v_i connects to a datacenter), otherwise, $x_i = 0$. So, the number of used datacenters N_{DC} (i.e., the number of DC-nodes.) can be calculated by

$$N_{DC} = \sum_{i=1}^{N_V} x_i.$$

Also, we have to look for the optimal deployment of VNFs on the DC-nodes such that the VNFs deployment on DC-nodes is as most balanced as possible. Let

$$y = (y_1, y_2, \cdots, y_k, \cdots, y_{N_T})$$

denote the scheme of VNF deployment for all VNF-SCs, where

$$y_k = (y_{im}^k)_{N_{DC}\times M_k}$$

is an $N_{DC} \times M_k$ matrix and represents the scheme of VNF deployment for task T_k, where $y_{im}^k = 1$ if and only if VNF_{k_m} in T_k is realized on DC-node v_i, otherwise, $y_{im}^k = 0$.

The number of VNFs assigned to the i-th DC-node in all tasks can be calculated by

$$N_{vnf}^i = \sum_{k=1}^{N_T}\sum_{m=1}^{M_k} y_{im}^k \tag{1}$$

The total number of VNFs in all VNF-SCs can be calculated by

$$N_{vnf}^T = \sum_{i=1}^{N_{DC}} N_{vnf}^i = \sum_{k=1}^{N_T} M_k \tag{2}$$

Thus the average number of VNFs assigned to one DC-node is $\frac{N_{vnf}^T}{N_{DC}}$. To make the balanced deployment of VNFs in all DC-nodes, the number of VNFs assigned to each DC-node should be as close to the average number as possible, i.e., the variance of these numbers, denoted by V_{ar}, should be as small as possible. It can be calculated by

$$\begin{aligned} V_{ar} &= \frac{1}{N_{DC}}\sum_{i=1}^{N_{DC}}\left(N_{vnf}^i - \frac{N_{vnf}^T}{N_{DC}}\right)^2 \\ &= \frac{1}{N_{DC}^3}\sum_{i=1}^{N_{DC}}\left(N_{DC}N_{vnf}^i - N_{vnf}^T\right)^2 \end{aligned} \tag{3}$$

So the second objective of the leader is to minimize V_{ar}. Now we have two objectives of leader: minimize N_{DC} and minimize V_{ar}. To make the problem easier, we integrate these two objectives into one objective. Considering that N_{DC} and V_{ar} are two different measures and on different orders of magnitudes, we can normalize the them, respectively. Note that $N_{DC} \leq N_V$. We can normalize N_{DC} by N_{DC}/N_V. Then we have $0 \leq N_{DC}/N_V \leq 1$.

Also note that when all VNFs in all VNF-SCs are assigned to one DC-node, V_{ar} will arrive at the maximum value denoted by V_{AR}. We can normalize V_{ar} by V_{ar}/V_{AR}, where V_{AR} can be easily calculated by

$$\begin{aligned} V_{AR} &= \frac{(N_{DC}-1)\frac{(N_{vnf}^T)^2}{N_{DC}^2} + \left(N_{vnf}^T - \frac{N_{vnf}^T}{N_{DC}}\right)^2}{N_{DC}} \\ &= \frac{\frac{(N_{vnf}^T)^2}{N_{DC}^2}\left((N_{DC}-1)+(N_{DC}-1)^2\right)}{N_{DC}} \\ &= \frac{(N_{vnf}^T)^2\left(N_{DC}^2 - N_{DC}\right)}{N_{DC}^3} \\ &= \frac{(N_{vnf}^T)^2\left(N_{DC}-1\right)}{N_{DC}^2} \end{aligned} \tag{4}$$

Thus,

$$\begin{aligned} \frac{V_{ar}}{V_{AR}} &= \frac{\frac{1}{N_{DC}^3}\sum_{i=1}^{N_{DC}}\left(N_{DC}N_{vnf}^i - N_{vnf}^T\right)^2}{\frac{(N_{vnf}^T)^2(N_{DC}-1)}{N_{DC}^2}} \\ &= \frac{\sum_{i=1}^{N_{DC}}\left(N_{DC}N_{vnf}^i - N_{vnf}^T\right)^2}{N_{DC}(N_{DC}-1)(N_{vnf}^T)^2} \\ &= \frac{N_{DC}\sum_{i=1}^{N_{DC}}(N_{vnf}^i)^2 - 2N_{vnf}^T\sum_{i=1}^{N_{DC}}N_{vnf}^i + (N_{vnf}^T)^2}{(N_{DC}-1)(N_{vnf}^T)^2} \\ &= \frac{N_{DC}\sum_{i=1}^{N_{DC}}(N_{vnf}^i)^2 - (N_{vnf}^T)^2}{(N_{DC}-1)(N_{vnf}^T)^2} \leq 1 \end{aligned} \tag{5}$$

Now we integrate the two objectives into one to be minimized as follows

$$\min_x f(x,y) = \min_x \left\{\left(\alpha\frac{N_{DC}}{N_V} + \beta\frac{V_{ar}}{V_{AR}}\right)\right\} \tag{6}$$

where α and β are two weights to adjust the importance of the two objectives with $0 \leq \alpha, \beta \leq 1, \alpha + \beta = 1$.

The leader's decision should be made under some conditions. These conditions constitute the constraints of the leader level problem as follows:

Constraint (a): The number of DC-nodes should not be greater than the number of nodes in EONs and not be less than n_{dc}, where n_{dc} is threshold of the minimum number of the datacenters, and is given by the experts in advance. That is

$$n_{dc} \leq N_{DC} = \sum_{i=1}^{N_V} x_i \leq N_V, \tag{7}$$

2) FOLLOWER'S OBJECTIVE

The smaller the maximum index of used frequency slots is, the smaller the bandwidth used in the network will be. Thus, one objective of the follower's problem is to minimize maximum index of used frequency slots N_F^m. Let

$$z = (z_1, z_2, \cdots, z_k, \cdots, z_{N_T})$$

be the scheme of path selection for all VNF-SCs, where

$$z_k = (z_{ij}^k)_{N_V\times N_V}$$

is an $N_V \times N_V$ matrix and represents the scheme of routing for VNF-SC T_k, where $z_{ij}^k = 1$ if and only if T_k occupies the l_{ij}, otherwise, $z_{ij}^k = 0$. Let

$$w = (w_1, w_2, \cdots, w_{N_T})$$

be the scheme of spectrum assignment for all VNF-SCs, where

$$w_k = (w_{ij}^{ku})_{N_V\times N_V}$$

is an $N_V \times N_V \times N_F$ matrix and represents the scheme of spectrum assignment for VNF-SC T_k, where $w_{ij}^{ku} = 1$ if and only if T_k occupies u-th frequency slot f_u on the l_{ij}, otherwise, $w_{ij}^{ku} = 0$. So, the maximum index of used frequency slots on l_{ij} of T_k occupied is

$$z_{ij}^k(w_{ij}^k + D_{ij}^k + G_F - 1)$$

where w_{ij}^k is the minimum index of used frequency slots on l_{ij} of T_k occupied, i.e., $w_{ij}^k = \arg min_{w_{ij}^{ku}=1}\{u\}$. G_F is the number of guaranteed frequency slots. D_{ij}^k is the number of used frequency slots on l_{ij} of T_k. In our work, we assume that each frequency slot has the same bandwidth C_{fs}, and the capacity of a frequency slot is $ML \times C_{fs}$, where ML is the bits per symbol in a specific modulation level. ML can be assigned as $1, 2, 3$ and 4 for different modulation level of BPSK, QPSK, 8QAM and 16QAM. If the modulation level of connection request r_k is denoted by ML_k, and b_k^m is the the number of frequency slots occupied by the changed data after VNF_{k_m} realized on link l_{ij}. D_{ij}^k can be calculated by

$$D_{ij}^k = \left\lceil \frac{b_k^m}{ML_k \times C_{fs}} \right\rceil. \tag{8}$$

The maximum index of used frequency slots on l_{ij} can be calculate by

$$\max_{T_k \in T} \left\{ z_{ij}^k(w_{ij}^k + D_{ij}^k + G_F - 1) \right\}$$

The maximum index of used frequency slots N_m^u is the maximum of used frequency slots of all links and can be calculated by

$$N_F^m = \max_{l_{ij} \in E} \left\{ \max_{T_k \in T} \left\{ z_{ij}^k(w_{ij}^k + D_{ij}^k + G_F - 1) \right\} \right\} \tag{9}$$

The number of used frequency slots of T_k on the links is

$$D_k = \sum_{l_{ij} \in E, z_{ij}^k = 1} D_{ij}^k \tag{10}$$

The number of used frequency slots of all the VNF-SC can be calculated by

$$N_F^u = \sum_{T_k \in T} D_k = \sum_{T_k \in T} \left(\sum_{l_{ij} \in E, z_{ij}^k = 1} D_{ij}^k \right) \tag{11}$$

So the second objective of the follower is to minimize N_F^u. Now we have two objectives of leader: minimize N_F^m and minimize N_F^u. Similar to the objective of leader, to make the problem easier, we integrate these two objectives into one objective. In addition, N_F^m and N_F^u are two different measures and on different orders of magnitudes, we also can normalize the them, respectively. Note that $N_F^m \le N_F$. We can normalize N_F^m by N_F^m / N_F. Thus, we have $0 \le N_F^m / N_F \le 1$. Since there are $N_F\ timesN_E$ frequency slots in this network, we have $N_F^u \le N_F \times N_E$. We can normalize N_F^u by N_F^u by $N_F^u / (N_F \times N_E)$. Then we have $0 \le N_F^u / (N_F \times N_E) \le 1$.

Now we integrate the two objectives into one to be minimized as follows

$$\min_{y,z,w} g(x, y, z, w) = \min_{y,z,w} \left\{ \gamma \frac{N_F^m}{N_F} + \delta \frac{N_F^u}{N_F \times N_E} \right\} \tag{12}$$

where γ and δ two weights to adjust the importance of the two objectives, and we have $0 \le \gamma, \delta \le 1, \gamma + \delta = 1$.

The follower's decision should be made under some conditions. These conditions constitute the constraints of the follower level problem as follows:

Constraint (b): For each VNF-SC $T_k (\forall T_k \in T)$, the links of T_k selection should include one or more DC-nodes.

$$\sum_{z_{ij}^k = 1} x_i \ge 1, \quad \forall T_k \in T \tag{13}$$

Constraint (c): Each VNF of $T_k (\forall T_k \in T)$ required only can be deployed in one DC-node. We can express this constraint by

$$\sum_{z_{ij}^k = 1} x_i y_{im}^k = 1, \quad \forall T_k \in T, \ VNF_{k_m} \in VNF_k^T \tag{14}$$

Constraint (d): For $T_k (\forall T_k \in T)$, the start index of occupied frequency slots on different links of a path must be identical. This can be given by

$$z_{ij}^k w_{ij}^k = z_{i'j'}^k w_{i'j'}^k, \quad \forall T_k \in T \tag{15}$$

where l_{ij} and $l_{i'j'}$ are the different links in the network.

Constraint (e): For $T_k (\forall T_k \in T)$, we must assign several consecutive frequency slots to it. That is

$$\prod_{u = w_{ij}^k}^{w_{ij}^k + D_{ij}^k + G_F - 1} w_{ij}^{ku} = 1, \quad \forall T_k \in T, \ z_{ij}^k = 1 \tag{16}$$

Constraint (f): For any two VNF-SCs T_k and $T_{k'}$ which occupy the same link l_{ij}, if the start frequency slot index of T_k is smaller than that of $T_{k'}$, this case is denoted by $T_k \prec T_{k'}$. Then these two VNF-SCs should satisfy

$$z_{ij}^k w_{ij}^k + B_k + G_F \le z_{ij}^{k'} w_{ij}^{k'}, \quad \forall T_k \prec T_{k'} \tag{17}$$

IV. PROPOSED ALGORITHM

The problem of datacenter placement, path selection and VNF deployment for VNF-SC in EONs is a hardest combinatorial optimization problems. The existing algorithms cannot be applied directly, and are necessary to make some improvements or revisions. To solve the bi-level programming model established, we propose an efficient algorithm and denote it as BiHMA (Bilevel Hybrid Memetic Algorithm).

A. ENCODING

There are four necessary steps to solve the problem: 1) determining the scheme of the datacenter placement; 2) path selection (selecting a path for each VNF-SC); 3) VNF deployment; 4) spectrum assignment. In our work, we use greedy strategy to determine the VNF deployment scheme, and use first fit strategy to assign spectra [4], [17]. Thus, we do not encode in the step of spectrum assignment. So it is necessary and reasonable to use two populations, i.e., datacenter placement population, path selection population.

In datacenter placement population, each individual presents a datacenter placement scheme. We assume that $x = (x_1, x_2, \cdots, x_{N_V})$ is an individual in datacenter placement population. We have $x_i = 1 (1 \le i \le N_V)$ if and only if a datacenter connect to node v_i.

Similar to datacenter placement population, each individual in path selection population presents a path selection scheme for all the VNF-SC. $Q_k = \{Q_k^1, Q_k^2, \cdots, Q_k^q, \cdots, Q_k^{N_k^P}\}$ denotes the candidate paths set of VNF-SC T_k which is calculated by K-Dijkstra allocation in advance, where N_k^P is the number of the candidate paths and Q_k^q is the q-th path. We assume that $h = (h_1, h_2, \cdots, h_{N_T})$ is an individual in path selection population. $h_k = q$ if and only if T_k occupies the path Q_k^q, and we have $z_{ij}^k = 1$ for $\forall l_{ij}$ in path Q_k^q.

1) POPULATION INITIALIZATION

In routing population initiation algorithm, uniform design [16], [18], [24], [26], [36], [43] is used to generate the routing individuals. The two Algorithms have an advantage that the individuals in initialized virtual nodes mapping population and routing population are all the feasible solutions. To understand Algorithm of routing population initiation clearly, we first introduce the uniform design method.

Overview of Uniform Design:

To generate points to be uniformly distributed on the experimental domain, uniform design method was developed [16], [18], [24], [26], [36]. It generates a small number of the uniformly distributed representative points in a domain by using a uniform array $U(S, H) = [U_{i,j}]_{H \times S}$, where $U_{i,j}$ denotes the level of the j-th factor in the i-th combination with the j-th factor representing the j-th variable and its level being its value [15].

To construct uniform design array, many methods are presented [16], [18], [24], [42]. Not only simple but also efficient method proposed in [24]. Firstly, we construct a hypercube over an S-dimensional space:

$$C^S = \{(c_1, c_2, \cdots, c_S) | a_i \le c_i \le b_i, i = 1, 2, \cdots, S\}$$

where a_i and b_i are the lower and upper bounds of the i-th factor (i.e., i-th variable), respectively. Then, a hyper-rectangle is formed between a_i and d_i as follows:

$$C(d) = \{(c_1, c_2, \cdots, c_S) | a_i \le c_i \le d_i, i = 1, 2, \cdots, S\} \subset C^S$$

Finally, H uniformly distributed points are selected randomly from C^S. Assuming that $H(d)$ is the number of points fallen into the hyper-rectangle $C(d)$, and the fraction of points in $C(d)$ is $H(d)/H$. As the volume of hypercube C^S is $\prod_{i=1}^S (b_i - a_i)$, so the volume of $C(d)$ is $\prod_{i=1}^S (d_i - a_i)$.

Algorithm 1 Crossover Operator for the Datacenter Placement Individual

Input: Individual x in the population, minimum number of datacenters n_{dc}

Output: Offspring x^c obtained by crossover operator

1 Two individuals x' and x'' are selected randomly;
2 Let $A = x^T x'$ and $C = x''^T x'$, $x^A = diag(A), x^C = diag(C)$;
3 **for** $i = 1$ *to* N_V **do**
4   **if** $rand() \le 0.5$ **then**
5     $x_i^c = x_i^A \bigvee x_i^C$;
6   **else**
7     $x_i^c = x_i^A \bigwedge x_i^C$;
8   **end**
9 **end**
10 **if** $diag((x^c)^T(x^c)) \le n_{dc}$ **then**
11   Let $n'_{dc} = \sum_{i=1}^{N_V} x_i^c$;
12   An integer n is generated in $[n_{dc}, N_V]$, $n' = n - n'_{dc}$;
13   $Pos = [ind_1, ind_2, \cdots, ind_{N_V - n'_{dc}}]$ denotes a index set that element is 0 in x^c;
14   Generate a permutation of Pos randomly, and denote is as Per, $x_{Per(1:n')}^c = 1$;
15 **end**

The H uniform distributed points in C^S should minimize

$$\sup_{x \in C^S} \left\{ \frac{H(d)}{H} - \frac{\prod_{i=1}^{S} (d_i - a_i)}{\prod_{i=1}^{S} (b_i - a_i)} \right\} \tag{18}$$

Hence, we can map these H points in C^S to the problem domain with S factors and χ levels uniformly, where H is an odd and $H > S$. It has been proved that $U_{i,j}$ can be given by [16], [18], [24]:

$$U_{i,j} = (i\sigma^{j-1} \mod \chi) + 1 \tag{19}$$

where σ is a constant related to the number of factors S and level χ. The H sample points scattered uniformly in the hypercube can be selected.

B. CROSSOVER OPERATORS

Since two different populations exist, different crossover operators are presented. Algorithm 1 is used to generate offspring for the datacenter placement individuals. As shown in line 1, two individual are selected randomly in datacenter placement population. Line 2 to line 9 are used to generate a new individual. Since the new individual may be a infeasible solution, line 10 to line 15 are used to modified the infeasible solution as a feasible solution. Offspring for path selection individuals are generated by Algorithm 2. As shown in line 1, a individual are selected randomly in path selection population. Line 2 to line 5 are used to generate a new individual. Since the new individual always a feasible solution, there is

Algorithm 2 Crossover Operator for the Path Selection Individual

Input: Individual h in the population
Output: Offspring h^c obtained by crossover operator
1 An individual $h' = (h'_1, h'_2, \cdots, h'_{N_T})$ is selected in the population;
2 Let $A = h^T h'$, and $C = diag(A)$
3 **for** $k = 1$ *to* N_T **do**
4 $\quad h_k = mod(C_k, N_k^P) + 1$;
5 **end**

no need to modify the individual. These two crossover operators have an advantage that the routing offspring obtained by it are all the feasible solutions.

C. MUTATION OPERATORS

Similar to crossover operator, there are two different mutation operators in proposed algorithm, and two mutation operators designed are shown in Algorithm 3 and Algorithm 4, respectively. As shown in algorithm 3, line 1 to 3 generate a new individual by reversing the sequence of the individual. To improve the search ability of the operator, line 4 to 5 generate a new individual by applying linear conversion. Line 6 to line 12 are used to modified the infeasible solution as a feasible solution. Algorithm 4 is the mutation operator for path selection population. Line to line are used to generate new individual, it can improve the diversity of the individuals.

Algorithm 3 Mutation Operator for the Datacenter Placement Individual

Input: Individual x in the population
Output: Offspring x^m obtained by mutation operator
1 **for** $i = 1$ *to* $\lfloor N_V/2 \rfloor$ **do**
2 $\quad x'_i = x_{N_V - i}$;
3 **end**
4 An integer p is generated randomly in $[1, N_V]$;
5 Let $A = x^T x'$, and $x^m = A(p, :)$;
6 **if** $\sum_{i=1}^{N_V} x_i^m < n_{dc}$ **then**
7 $\quad$ Let $n'_{dc} = \sum_{i=1}^{N_V} x_i^m$;
8 $\quad$ An integer n is generated in $[n_{dc}, N_V]$, $n' = n - n'_{dc}$;
9 $\quad$ $Pos = [ind_1, ind_2, \cdots, ind_{N_V - n'_{dc}}]$ denotes a index set that element is 0 in x^m;
10 $\quad$ Generate a permutation of Pos randomly, and denote is as Per;
11 $\quad$ $x^m_{Per(1:n')} = 1$;
12 **end**

Algorithm 4 Mutation Operator for the Path Selection Individual

Input: Individual h in the population
Output: Offspring h' obtained by mutation operator
1 **for** $k = 1$ *to* N_T **do**
2 $\quad h'_k = N_k^P - h_k$;
3 **end**
4 Let $A = h^T h'$, and $C = diag(A)$
5 **for** $k = 1$ *to* N_T **do**
6 $\quad h'_k = mod(C_k, N_k^P) + 1$;
7 **end**

by changing the value of gene. Line 7 to line 15 are used to modified the infeasible solution as a feasible solution. Similar to the local search operator for datacenter placement individual, A new path selection individual is generate by changing the value of gene. Then, the individual is modified as a feasible solution.

Algorithm 5 Local Search Operator for the Datacenter Placement Individual

Input: Individual x in datacenter placement population
Output: Offspring x^l obtained by local search operator
1 $x^l = x$, $N_{DC} = \sum_{i=1}^{N_V} x_i$;
2 **if** $N_{DC} = n_{dc}$ **then**
3 $\quad$ $Pos = [ind_1, ind_2, \cdots, ind_{N_V - N_{DC}}]$ denotes a index set that element is 0 in x;
4 $\quad$ Generate a permutation of Pos randomly, and denote is as Per;
5 $\quad$ $num = randint(N_V - N_{DC})$; $x^l_{Per(1:num)} = 1$;
6 **else**
7 $\quad$ **if** $rand() \leq 0.5$ **then**
8 $\quad\quad$ $Pos = [ind_1, ind_2, \cdots, ind_{N_V - N_{DC}}]$ denotes a index set that element is 0 in x;
9 $\quad\quad$ Generate a permutation of Pos randomly, and denote is as Per;
10 $\quad\quad$ $num = randint(N_V - N_{DC})$; $x^l_{Per(1:num)} = 1$;
11 $\quad$ **else**
12 $\quad\quad$ $Pos = [ind_1, ind_2, \cdots, ind_{N_{DC}}]$ denotes a index set that element is 1 in x;
13 $\quad\quad$ Generate a permutation of Pos randomly, and denote is as Per;
14 $\quad\quad$ $num = randint(N_{DC} - n_{dc})$; $x^l_{Per(1:num)} = 0$;
15 $\quad$ **end**
16 **end**

D. LOCAL SEARCH

Two local search operators for datacenter placement individual and path selection individual are shown in Algorithm 5 and Algorithm 6. Algorithm 5 is the local search operator for datacenter placement individual. A new individual is generate

V. EXPERIMENTS AND ANALYSIS

To demonstrate the effectiveness and efficiency of the proposed algorithm, several experiments are conducted on two widely used networks. In section V-A, the parameters used in the algorithms will be given. Experimental results are

Algorithm 6 Local Search for the Path Selection Individual

Input: Individual h in path selection population
Output: Offspring h^l obtained by local search operator

```
h^l = h; num = randint(N_V);
if rand() ≤ 0.5 then
    for i = 1 to N_V do
        h^l_{mod(i+num,N_V)+1} = h_i;
    end
else
    for i = 1 to N_V do
        h^l_{mod(i+N_V−num,N_V)+1} = h_i;
    end
end
```

presented in section V-B. Finally, experimental results are analyzed in section V-C.

A. PARAMETERS SETTING

In the experiments, two widely used networks are used, i.e., NSFNET with 14 nodes and 21 links and US Backbone with 27 nodes and 44 links, respectively [12], [34]. We assume that each frequency slot is 12.5 GHz, i.e., $C_{fs} =$ 12.5GHz. There are $N_{vnf} = 8$ VNFs, and each VNF-SC has 1-5 VNFs to realize. Each VNF-SC requires frequency slots satisfy uniform distribution in [5, 10], and Each link has 1000 frequency slots, i.e., $N_F = 1000$. In proposed memetic algorithm (we denote it as BiHMA), following parameters are chosen: population size $P_s = 100$, crossover probability $p_c =$ 0.8, mutation probability $p_m = 0.2$, maximum iterations $G_{max} = 1000$, number of elites is 10.

B. EXPERIMENTAL RESULTS

By our knowledge, there is no existing algorithm focusing on the problem of datacenter placement, we compare the proposed algorithm with other two algorithms, which investigate path selection and VNF deployment for VNF-SC in EONs. The one is modified by the algorithm proposed in literature [10](briefly LBA), and the other one is LF-LBA, which includes least fist strategy and LBA algorithm. In this way, we can make the comparisons between the proposed algorithms and these two efficient algorithms. In addition, to demonstrate the proposed BiHMA can solve the bi-level optimization model effectively, two recent algorithms, which denoted by PSO and DE, are selected to compared with BiHMA. Literature attempts to develop an efficient method based on particle swarm optimization (PSO) algorithm with swarm intelligence, and we denote the algorithm as PSO [21]. DE is proposed in literature [19], and use DE for solving bi-level programming problems with applications In the field of transportation planning. Fig.1 and Fig.2 show the leader's objective obtained in NSFNST topology and US Backbone topology when $\alpha = \beta = 0.5$. Fig.3 and Fig.4 show the leader's objective obtained in NSFNST topology and US Backbone topology when $\gamma = \delta = 0.5$. In each figure, there are four subfigures, and the experimental results are obtained with $N_{DC} = N_V/4$, $N_{DC} = N_V/2$, $N_{DC} = 3N_V/4$ and $N_{DC} = N_V$, respectively. In each subfigure, five groups experimental results are shown. In the five groups experiments, number of VNF-SCs are set as $N_T = \omega N_V(N_V - 1)$, and $\omega = 0.25, 0.5, 1, 2$ and 4, respectively.

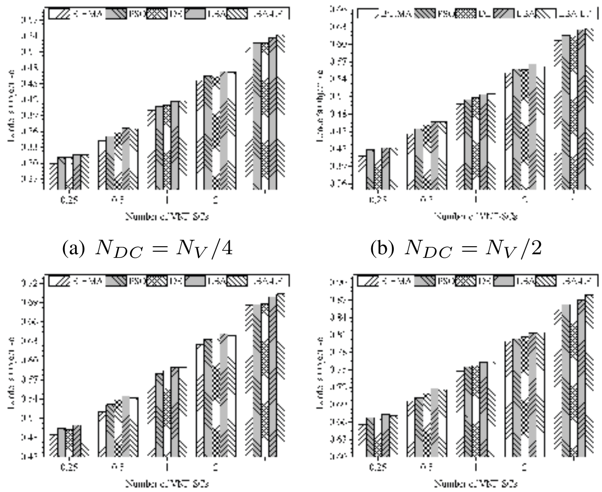

(a) $N_{DC} = N_V/4$ (b) $N_{DC} = N_V/2$ (c) $N_{DC} = 3N_V/4$ (d) $N_{DC} = N_V$

FIGURE 1. Leader's objective obtained in NSFNET topology.

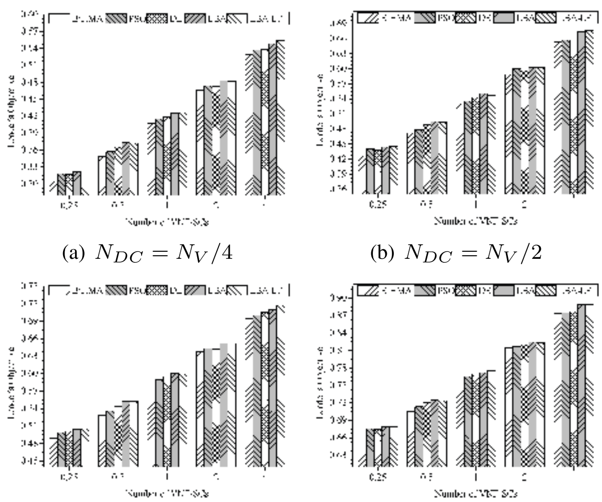
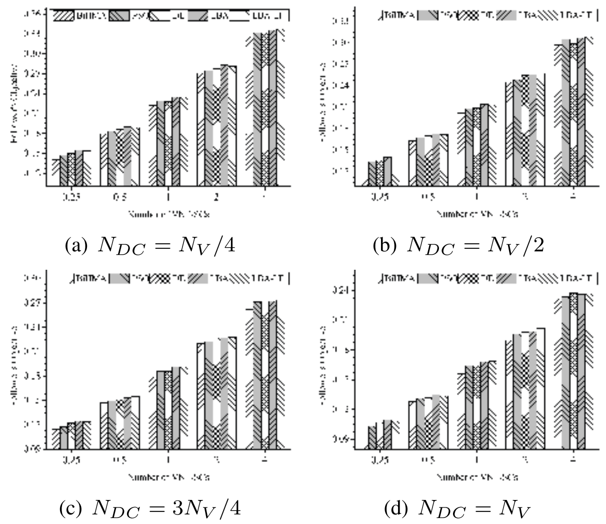

(a) $N_{DC} = N_V/4$ (b) $N_{DC} = N_V/2$ (c) $N_{DC} = 3N_V/4$ (d) $N_{DC} = N_V$

FIGURE 2. Leader's objective obtained in US Backbone topology.

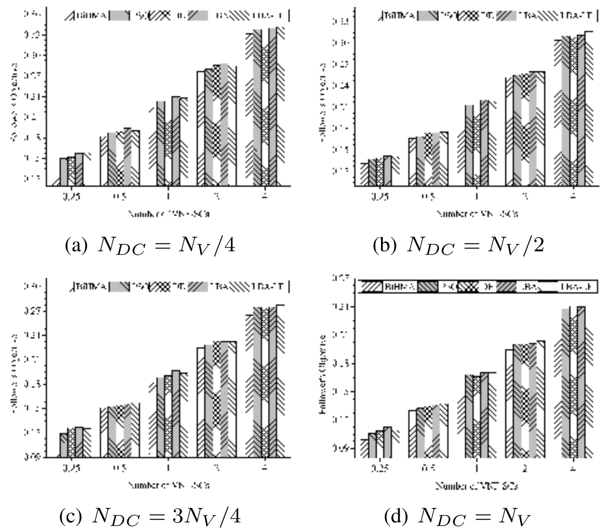
To demonstrate the high performance of our model and algorithm, we conduct another group experiments. In this experiment, number of datacenters and the location of all the datacenters are flexible. That is to say, our model and algorithm can determine the optimal number of datacenters and the location of the datacenters according to the VNF-SCs. Fig.5(a) and Fig.6(a) show the leader's objective obtained in NSFNST topology and US Backbone topology when $\alpha = \beta = 0.5$. Fig.5(b) and Fig.6(b) show the follower's objective obtained in NSFNST topology and US Backbone topology when $\gamma = \delta = 0.5$.



(a) $N_{DC} = N_V/4$ (b) $N_{DC} = N_V/2$

(c) $N_{DC} = 3N_V/4$ (d) $N_{DC} = N_V$

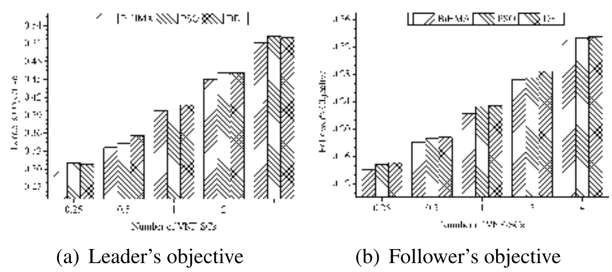
FIGURE 3. Follower's objective obtained in NSFNET topology.



(a) $N_{DC} = N_V/4$ (b) $N_{DC} = N_V/2$

(c) $N_{DC} = 3N_V/4$ (d) $N_{DC} = N_V$

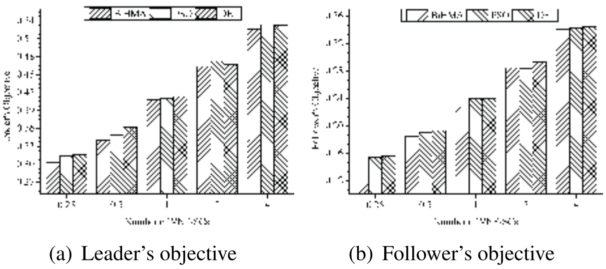
FIGURE 4. Follower's objective obtained in US Backbone topology.



(a) Leader's objective (b) Follower's objective

FIGURE 5. Objectives obtained in NSFNET topology.

To evaluate the stability of the proposed algorithm and the compared algorithm, we give the statistical results (Mean and Standard Deviation) in the two groups experiments with the different experimental scenes. Since LBA and LF-LBA are two deterministic algorithms, we not give the statistical results obtained by these two algorithms. Table 1 shows the mean and standard deviation results of the leader's objective on the two network topologies with different scenes in the first group experiments. The mean and standard deviation results of follower's objective on the two network topologies with different scenes in the first group experiments are shown in Table 2. We give the mean and standard deviation results of the leader's and Follower' objective of the second group experiments in Table 3.



(a) Leader's objective (b) Follower's objective

FIGURE 6. Objectives obtained in US Backbone topology.

TABLE 1. Statistical results (Mean and Standard Deviation) of the Leader's objective in first experiment instance.

	N	ω	BiHMA	PSO	DE
NSFNET	0.25	0.25	0.30493 (1.50E-2)	0.31767 (1.68E-2)	0.31706 (1.57E-2)
		0.5	0.34879 (2.69E-2)	0.35729 (3.52E-2)	0.36544 (3.81E-2)
		1	0.40741 (5.10E-2)	0.41477 (7.79E-2)	0.41816 (8.14E-2)
		2	0.46566 (8.47E-2)	0.47398 (1.07E-1)	0.47269 (1.54E-1)
		4	0.52931 (1.88E-1)	0.53711 (2.02E-1)	0.53763 (2.29E-1)
	0.5	0.25	0.42674 (3.55E-2)	0.44079 (3.41E-2)	0.43782 (3.94E-2)
		0.5	0.47216 (6.06E-2)	0.47834 (7.99E-2)	0.48822 (6.54E-2)
		1	0.53082 (8.85E-2)	0.53518 (9.79E-2)	0.54284 (1.14E-1)
		2	0.58842 (1.47E-1)	0.59987 (1.93E-1)	0.59633 (1.89E-1)
		4	0.65337 (1.88E-1)	0.65697 (2.02E-1)	0.65764 (2.29E-1)
	0.75	0.25	0.48934 (3.83E-2)	0.49846 (5.02E-2)	0.50089 (5.09E-2)
		0.5	0.52848 (6.06E-2)	0.53600 (7.99E-2)	0.54395 (6.14E-2)
		1	0.58396 (1.10E-1)	0.58932 (9.79E-2)	0.59507 (1.14E-1)
		2	0.63799 (1.47E-1)	0.64275 (1.07E-1)	0.64238 (1.54E-1)
		4	0.69484 (2.18E-1)	0.70005 (1.02E-1)	0.70513 (2.29E-1)
	1	0.25	0.66467 (3.26E-2)	0.67531 (3.48E-2)	0.67434 (3.94E-2)
		0.5	0.70510 (6.12E-2)	0.71349 (7.85E-2)	0.72043 (7.34E-2)
		1	0.75613 (8.39E-2)	0.76524 (9.79E-2)	0.76972 (1.01E-1)
		2	0.81404 (1.09E-1)	0.81696 (1.67E-1)	0.82008 (1.84E-1)
		4	0.87415 (1.67E-1)	0.87507 (2.03E-1)	0.87651 (2.21E-1)
US Backbone	0.25	0.25	0.30493 (2.58E-2)	0.31767 (2.94E-2)	0.31706 (3.05E-2)
		0.5	0.34879 (4.05E-2)	0.35729 (4.62E-2)	0.36544 (4.58E-2)
		1	0.40741 (5.82E-1)	0.41477 (6.34E-2)	0.41816 (6.57E-1)
		2	0.46566 (8.67E-2)	0.47398 (9.51E-2)	0.47269 (9.37E-2)
		4	0.52931 (1.27E-1)	0.53711 (1.85E-1)	0.53763 (2.07E-1)
	0.5	0.25	0.42674 (3.01E-2)	0.44079 (3.62E-2)	0.43782 (3.94E-2)
		0.5	0.47216 (6.06E-2)	0.47834 (7.99E-2)	0.48822 (6.14E-2)
		1	0.53082 (1.10E-1)	0.53518 (9.79E-2)	0.54284 (1.14E-1)
		2	0.58842 (1.47E-1)	0.59987 (1.07E-1)	0.59633 (1.54E-1)
		4	0.65337 (2.18E-1)	0.65697 (1.02E-1)	0.65764 (2.29E-1)
	0.75	0.25	0.48934 (2.94E-2)	0.49846 (3.65E-2)	0.50089 (3.57E-2)
		0.5	0.52848 (6.06E-2)	0.53600 (7.99E-2)	0.54395 (7.14E-2)
		1	0.58396 (8.10E-2)	0.58932 (9.79E-2)	0.59507 (1.04E-1)
		2	0.63799 (1.47E-1)	0.64275 (1.87E-1)	0.64238 (1.94E-1)
		4	0.69484 (1.98E-1)	0.70005 (2.62E-1)	0.70513 (2.39E-1)
	1	0.25	0.66467 (3.15E-2)	0.67531 (3.81E-2)	0.67434 (3.94E-2)
		0.5	0.70510 (5.84E-2)	0.71349 (7.05E-2)	0.72043 (6.97E-2)
		1	0.75613 (8.57E-2)	0.76524 (9.79E-2)	0.76972 (9.14E-2)
		2	0.81404 (1.06E-1)	0.81696 (1.84E-1)	0.82008 (1.92E-1)
		4	0.87415 (1.76E-1)	0.87507 (2.13E-1)	0.87651 (2.34E-1)

C. EXPERIMENTAL ANALYSIS

In the first group experiments, the leader's objective obtained by BiHMA and four compared algorithms (LBA, LF-LBA, PSO and DE) are shown in Fig.1 and Fig.2. From the experimental results, we can see that the leader's objective obtained by BiHMA are much less than those obtained by

TABLE 2. Statistical results (Mean and Standard Deviation) of the Follower's objective in first experiment instance.

	N	ω	BiHMA	PSO	DE
NSFNET	0.25	0.25	0.13977 (1.89E-2)	0.14697 (2.37E-2)	0.15039 (2.19E-2)
		0.5	0.17946 (4.37E-2)	0.18297 (5.38E-2)	0.18585 (5.19E-2)
		1	0.22212 (6.18E-2)	0.22905 (7.21E-2)	0.22779 (7.67E-2)
		2	0.27054 (8.67E-2)	0.27423 (9.23E-2)	0.27783 (1.04E-1)
		4	0.32616 (1.16E-1)	0.33192 (1.76E-1)	0.33147 (1.90E-1)
	0.5	0.25	0.12849 (2.37E-2)	0.13411 (3.61E-2)	0.13546 (3.24E-2)
		0.5	0.16348 (4.21E-2)	0.16716 (5.16E-2)	0.17056 (5.34E-2)
		1	0.20222 (6.82E-2)	0.20796 (8.41E-2)	0.20917 (8.67E-2)
		2	0.24474 (9.54E-2)	0.24882 (1.12E-1)	0.25447 (1.09E-1)
		4	0.29673 (1.27E-1)	0.30529 (1.94E-1)	0.29898 (1.82E-1)
	0.75	0.25	0.11434 (3.07E-2)	0.11767 (3.94E-2)	0.12215 (4.01E-2)
		0.5	0.14677 (4.95E-2)	0.14910 (5.62E-2)	0.15098 (5.59E-2)
		1	0.17918 (6.98E-1)	0.18553 (7.56E-2)	0.18615 (7.84E-1)
		2	0.22021 (8.27E-1)	0.22203 (9.57E-1)	0.22674 (9.34E-1)
		4	0.26222 (1.67E-1)	0.27154 (1.98E-1)	0.26900 (1.86E-1)
	1	0.25	0.09801 (2.18E-2)	0.10316 (3.52E-2)	0.10726 (3.29E-2)
		0.5	0.12807 (3.81E-2)	0.13107 (4.73E-2)	0.13202 (4.91E-2)
		1	0.15567 (5.73E-2)	0.16408 (6.65E-2)	0.16389 (6.29E-2)
		2	0.18976 (7.91E-2)	0.19592 (9.64E-2)	0.19809 (9.72E-2)
		4	0.23171 (9.92E-2)	0.23334 (1.67E-1)	0.23733 (1.89E-1)
US Backbone	0.25	0.25	0.14158 (1.98E-2)	0.15002 (2.41E-2)	0.15227 (2.34E-2)
		0.5	0.18193 (3.72E-2)	0.18734 (4.58E-2)	0.18953 (4.13E-2)
		1	0.22453 (5.19E-2)	0.23363 (6.85E-2)	0.23395 (7.57E-2)
		2	0.27668 (7.34E-2)	0.27991 (9.58E-2)	0.28546 (9.07E-2)
		4	0.33126 (9.19E-2)	0.33819 (1.27E-1)	0.33658 (1.53E-1)
	0.5	0.25	0.13137 (3.57E-2)	0.13743 (4.78E-2)	0.13914 (4.53E-2)
		0.5	0.16699 (5.10E-2)	0.16904 (6.48E-2)	0.17495 (6.62E-2)
		1	0.20445 (7.51E-2)	0.21362 (8.16E-2)	0.21341 (8.24E-2)
		2	0.25175 (9.53E-2)	0.25596 (1.02E-1)	0.25792 (1.23E-1)
		4	0.30402 (1.68E-1)	0.31035 (1.85E-1)	0.30835 (1.76E-1)
	0.75	0.25	0.11676 (4.01E-2)	0.11926 (4.18E-2)	0.12579 (4.24E-2)
		0.5	0.15074 (5.84E-2)	0.15257 (6.49E-2)	0.15412 (6.34E-2)
		1	0.18222 (7.29E-2)	0.18861 (9.16E-2)	0.19064 (9.62E-2)
		2	0.22438 (9.87E-2)	0.22823 (1.28E-1)	0.23354 (1.32E-1)
		4	0.26513 (1.85E-1)	0.27490 (2.03E-1)	0.27324 (1.92E-1)
	1	0.25	0.09933 (4.34E-2)	0.10607 (4.36E-2)	0.10873 (4.91E-2)
		0.5	0.12989 (5.98E-2)	0.13265 (6.49E-2)	0.13492 (6.31E-2)
		1	0.16004 (8.13E-2)	0.16817 (9.57E-2)	0.16651 (9.96E-2)
		2	0.19422 (1.37E-1)	0.20076 (1.92E-1)	0.20060 (2.31E-1)
		4	0.23120 (1.85E-1)	0.23830 (2.52E-1)	0.24071 (2.49E-1)

TABLE 3. Statistical results (Mean and Standard Deviation) in second experiment instance.

		ω	BiHMA	PSO	DE
NSFNET	Leader	0.25	0.29616 (1.85E-2)	0.30986 (1.77E-2)	0.30789 (2.08E-2)
		0.5	0.33586 (4.05E-2)	0.34335 (4.92E-2)	0.35633 (4.83E-2)
		1	0.39803 (6.96E-2)	0.39890 (6.73E-2)	0.40756 (6.07E-2)
		2	0.45049 (8.26E-2)	0.46160 (9.14E-2)	0.46139 (9.21E-2)
		4	0.51160 (1.85E-1)	0.52240 (2.02E-1)	0.51958 (1.97E-1)
	Follower	0.25	0.14071 (2.84E-2)	0.14803 (3.67E-2)	0.15090 (3.21E-2)
		0.5	0.18032 (4.64E-2)	0.18587 (5.19E-2)	0.18758 (5.25E-2)
		1	0.22182 (6.84E-2)	0.23242 (7.61E-2)	0.23344 (7.92E-2)
		2	0.27149 (9.04E-2)	0.27490 (9.79E-2)	0.28431 (1.02E-1)
		4	0.32980 (1.68E-1)	0.33229 (2.47E-1)	0.33479 (2.08E-1)
US Backbone	Leader	0.25	0.30350 (2.36E-2)	0.31349 (3.04E-2)	0.31583 (3.41E-2)
		0.5	0.34080 (4.01E-2)	0.34930 (4.97E-2)	0.36251 (5.24E-2)
		1	0.40786 (7.28E-2)	0.40983 (7.61E-2)	0.41331 (7.93E-2)
		2	0.46374 (9.57E-1)	0.47254 (1.16E-1)	0.46680 (1.54E-1)
		4	0.52559 (1.61E-1)	0.53387 (1.99E-1)	0.53280 (2.04E-1)
	Follower	0.25	0.14314 (2.18E-2)	0.15357 (2.83E-2)	0.15574 (2.949E-2)
		0.5	0.18401 (3.89E-2)	0.18971 (4.53E-2)	0.19267 (4.71E-2)
		1	0.22717 (5.19E-2)	0.23934 (6.24E-2)	0.23935 (6.85E-2)
		2	0.28386 (8.24E-2)	0.28333 (9.61E-2)	0.29259 (9.78E-2)
		4	0.34082 (1.84E-1)	0.34237 (2.36E-1)	0.34387 (2.64E-1)

the four compared algorithms. Compared algorithms (LBA, LF-LBA) are not consider the datacenter placement and balanced VNF deployment, our proposed algorithm can determine the optimal datacenter placement scheme and VNF deployment scheme. So, the leader's objective obtained by BiHMA are much less than those obtained by these two compared algorithms. An efficient local search operator is designed, so the BiHMA can obtain a better schemes of datacenter placement and VNF deployment than the compared algorithms (PSO and DE). As shown in Fig.1 and Fig.2, leader's objective obtained by the proposed algorithms are 7.9%-10.5% less than those obtained by the four compared algorithms when the number of VNF-SCs is $0.25N_V(N_V-1)$, respectively. When the number of VNF-SCs is $4N_V(N_V-1)$, leader's objective obtained by the proposed algorithms are 12.5%-14.8% less than those obtained by the four compared algorithms. For the same number of datacenters, we can see that proposed algorithms can obtain a smaller leader's objective than the four compared algorithms with the increase of the number of VNF-SCs. That is to say, proposed algorithm can obtain a more balanced VNF deployment than the four compared algorithms with the increase of the number of VNF-SCs. For the same number of VNF-SCs, when number of data centers is $N_V/4$, the leader's objective obtained by the proposed algorithms are 4.3%-6.2% less than those obtained by the four compared algorithms. Leader's objective obtained by the proposed algorithms are 6.8%-9.1% less than those obtained by the four compared algorithms when number of data centers is N_V. That is to say, proposed algorithm can obtain a more balanced VNF deployment than the four compared algorithms with the increase of the number of data centers.

In the first group experiments, the follower's objective obtained by BiHMA and four compared algorithms are shown in Fig.3 and Fig.4. From the experimental results, we can see that the follower's objective obtained by BiHMA are much less than those obtained by the four compared algorithms. In proposed model, not only is the maximum index of used frequency slots considered, but also the number of frequency slots used is taken into account. In addition, proposed algorithm has a higher search ability than the four compared algorithms and can find the optimal solution. So, the follower's objective obtained by BiHMA are much less than those obtained by the four compared algorithms. As shown in Fig.3 and Fig.4, follower's objective obtained by the proposed algorithms are 4.5%-7.2% less than those obtained by algorithms LBA and LF-LBA, respectively when the number of VNF-SCs is $0.25N_V(N_V-1)$. When the number of VNF-SCs is $4N_V(N_V-1)$, follower's objective obtained by the proposed algorithms are 11.2%-15.7% less than those obtained by algorithms LBA and LF-LBA, respectively. For the same number of datacenters, we can see that proposed algorithms can obtain a smaller follower's objective than the four compared algorithms with the increase of the number of VNF-SCs. That is to say, proposed algorithm can obtain a smaller maximum index of used frequency slots and save more frequency slots used than the four compared algorithms with the increase of the number of VNF-SCs. For the same number of VNF-SCs and save network topology, we can find that the follower's objective is decreased with the increase of the number of datacenters. When the number of datacenters is $N_V/4$, it will result that some links occupied by a large number of VNF-SCs became key links. So, the frequency slots used are imbalanced on different links. The VNF-SCs are deployed on different links balanced when the number of datacenters is N_V. So, the the maximum index of used frequency slots and the number of frequency slots used are small.

In the second group experiments, the number of datacenters is not fixed in advance, and only the information of VNF-SCs are given to determine the optimal datacenter placement scheme, path selection and VNF deployment scheme. In this proposed algorithm, uniform design is used to generate the population, it can help to improve the homogeneity of the individuals. In addition, well-designed and tailor-made crossover and mutation operators can generate some better individuals. So, proposed algorithm has a higher search ability than the two compared algorithms (PSO and DE) and can find the optimal solution. As shown in the experimental results, we can sea that BiHMA can obtain the better leader's objective and follower's objective than the compared algorithms.

Table 1, Table 2 and Table 3 show the statistical results (mean and standard deviation) on the two network topologies with two instances. From the statistical results, we can see that BiHMA is significant better than the compared algorithms. Not only the statistical results of mean but also the statistical results of standard deviation are smaller than the compared algorithms. That is to say, the proposed algorithm can obtain the better results and stability results than the compared algorithms. The efficient crossover, and mutation operator in BiHMA, which can serch the optimal schemes of datacenter placement and path selection, are designed. So BiHMA can obtain a better results (Mean) than the compared algorithms (PSO and DE). In addition, BiHMA includes the efficient local search operator,which can search the optimal number and location of the datacenters, optimal path selection. Therefore, BiHMA can obtain a better results (standard deviation) than the compared algorithms (PSO and DE).

VI. CONCLUSION

In this paper, we investigate a network planning problem in Inter-DC EONs by considering all these factors, i.e.,we should determine not only the optimal routing and VNF deployment scheme for VNF-SC, but also the optimal number and location of datacenters. A bi-level programming model, which includes a leader' objective and a follower's objective, is established to tackle this challenging problem. To solve the whole model effectively, we proposed an efficient bi-level hybrid memetic algorithm with tailor-made crossover operators and mutation operators. To demonstrate reasonable of the model and high performance of the designed algorithm, a series of experiments are conducted with several different scenes. Experimental results demonstrate that proposed algorithm have a higher performance than the compared algorithms.

REFERENCES

[1] B. Addis, D. Belabed, M. Bouet, and S. Secci, "Virtual network functions placement and routing optimization," in *Proc. IEEE 4th Int. Conf. Cloud Netw. (CloudNet)*, Oct. 2015, pp. 171–177.

[2] S. Ahvar, H. P. Phyu, S. M. Buddhacharya, E. Ahvar, N. Crespi, and R. Glitho, "CCVP: Cost-efficient centrality-based VNF placement and chaining algorithm for network service provisioning," in *Proc. IEEE Conf. Netw. Softwarization (NetSoft)*, Jul. 2017, pp. 1–9.

[3] M. T. Beck and J. F. Botero, "Scalable and coordinated allocation of service function chains," *Comput. Commun.*, vol. 102, pp. 78–88, Apr. 2017.

[4] A. Bocoi, M. Schuster, F. Rambach, M. Kiese, C.-A. Bunge, and B. Spinnler, "Reach-dependent capacity in optical networks enabled by OFDM," in *Proc. Conf. Opt. Fiber Commun.—Incudes Post Deadline Papers*, 2009, pp. 1–3.

[5] M. A. Cavalcante, H. A. Pereira, D. A. R. Chaves, and R. C. Almeida, "Evolutionary multiobjective strategy for regenerator placement in elastic optical networks," *IEEE Trans. Commun.*, vol. 66, no. 8, pp. 3583–3596, Aug. 2018.

[6] B. Chatras and F. F. Ozog, "Network functions virtualization: The portability challenge," *IEEE Netw.*, vol. 30, no. 4, pp. 4–8, Jul./Aug. 2016.

[7] X. Chen, Z. Zhu, J. Guo, S. Kang, R. Proietti, A. Castro, and S. J. B. Yoo, "Leveraging mixed-strategy gaming to realize incentive-driven vnf service chain provisioning in broker-based elastic optical inter-datacenter networks," *J. Opt. Commun. Netw.*, vol. 10, no. 2, pp. A232–A240, Feb. 2018.

[8] V. Eramo and F. G. Lavacca, "Optimizing the cloud resources, bandwidth and deployment costs in multi-providers network function virtualization environment," *IEEE Access*, vol. 7, pp. 46898–46916, 2019.

[9] V. Eramo and F. G. Lavacca, "Proposal and investigation of a reconfiguration cost aware policy for resource allocation in multi-provider nfv infrastructures interconnected by elastic optical networks," *J. Lightw. Technol.*, vol. 37, no. 16, pp. 4098–4114, Aug. 2019.

[10] W. Fang, M. Zeng, X. Liu, W. Lu, 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, Aug. 2016.

[11] M. D. Firoozjaei, J. P. Jeong, H. Ko, and H. Kim, "Security challenges with network functions virtualization," *Future Gener. Comput. Syst.*, vol. 67, pp. 315–324, Feb. 2017.

[12] L. Gong, X. Zhou, X. Liu, W. Zhao, W. Lu, and Z. Zhu, "Efficient resource allocation for all-optical multicasting over spectrum-sliced elastic optical networks," *J. Opt. Commun. Netw.*, vol. 5, no. 8, pp. 836–847, Aug. 2013.

[13] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *IEEE Commun. Mag.*, vol. 53, no. 2, pp. 90–97, Feb. 2015.

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

[15] L. Jia, Y. Wang, and L. Fan, "Multiobjective bilevel optimization for production distribution planning problems using hybrid genetic algorithm," *Integr. Comput.-Aided Eng.*, vol. 21, no. 1, pp. 77–90, 2014.

[16] S. Jiang and Z. Cai, *A New Differential Evolution for Multiobjective Optimization by Uniform Design and Minimum Reduce Hypervolume*. Tokyo, Japan: Springer, 2010, pp. 199–208.

[17] M. Jinno, B. Kozicki, H. Takara, A. Watanabe, Y. Sone, Y. Sone, T. Tanaka, and A. Hirano, "Distance-adaptive spectrum resource allocation in spectrum-sliced elastic optical path network [Topics in optical communications]," *IEEE Commun. Mag.*, vol. 48, no. 8, pp. 138–145, Aug. 2010.

[18] K.-T. Fang, D. K. J. Lin, P. Winker, and Y. Zhang, "Uniform design: Theory and application," *Technometrics*, vol. 42, no. 3, pp. 237–248, 2000.

[19] A. Koh, "Solving transportation bi-level programs with differential evolution," in *Proc. IEEE Congr. Evol. Comput.*, Sep. 2007, pp. 2243–2250.

[20] R. J. Kuo and Y. S. Han, "A hybrid of genetic algorithm and particle swarm optimization for solving bi-level linear programming problem—A case study on supply chain model," *Appl. Math. Model.*, vol. 35, no. 8, pp. 3905–3917, 2011.

[21] R. J. Kuo and C. C. Huang, "Application of particle swarm optimization algorithm for solving bi-level linear programming problem," *Comput. Math. With Appl.*, vol. 58, no. 4, pp. 678–685, 2009.

[22] T.-W. Kuo, B.-H. Liou, K. C.-J. Lin, and M.-J. Tsai, "Deploying chains of virtual network functions: On the relation between link and server usage," in *Proc. IEEE INFOCOM 35th Annu. IEEE Int. Conf. Comput. Commun.*, Apr. 2016, pp. 1–9.

[23] A. Leivadeas, G. Kesidis, M. Falkner, and I. Lambadaris, "A graph partitioning game theoretical approach for the VNF service chaining problem," *IEEE Trans. Netw. Service Manag.*, vol. 14, no. 4, pp. 890–903, Dec. 2017.

[24] Y.-W. Leung and Y. Wang, "Multiobjective programming using uniform design and genetic algorithm," *IEEE Trans. Syst., Man, Cybern. C (Applications Reviews)*, vol. 30, no. 3, pp. 293–304, Aug. 2000.
[25] B. Li, W. Lu, S. Liu, and Z. Zhu, "Deep-learning-assisted network orchestration for on-demand and cost-effective vnf service chaining in inter-dc elastic optical networks," *J. Opt. Commun. Netw.*, vol. 10, no. 10, pp. D29–D41, Oct. 2018.
[26] W. Li, L. Liu, and W. Gong, "Multi-objective uniform design as a SVM model selection tool for face recognition," *Expert Syst. With Appl.*, vol. 38, no. 6, pp. 6689–6695, 2011.
[27] X. Li and C. Qian, "The virtual network function placement problem," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2015, pp. 69–70.
[28] X. Li, W. Li, X. Cai, and F. He, "A hybrid optimization approach for sustainable process planning and scheduling," *Integr. Comput.-Aided Eng.*, vol. 22, no. 4, pp. 311–326, 2015.
[29] M. Mechtri, C. Ghribi, and D. Zeghlache, "A scalable algorithm for the placement of service function chains," *IEEE Trans. Netw. Service Manag.*, vol. 13, no. 3, pp. 533–546, Sep. 2016.
[30] R. Mencía, M. R. Sierra, C. Mencía, and R. Varela, "Genetic algorithms for the scheduling problem with arbitrary precedence relations and skilled operators," *Integr. Comput.-Aided Eng.*, vol. 23, no. 3, pp. 269–285, 2016.
[31] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 236–262, 1st Quart., 2016.
[32] M. Moharrami, A. Fallahpour, H. Beyranvand, and J. A. Salehi, "Resource allocation and multicast routing in elastic optical networks," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2101–2113, May 2017.
[33] A. Muhammad, M. Fiorani, L. Wosinska, and J. Chen, "Joint optimization of resource allocation for elastic optical intra-datacenter network," *IEEE Commun. Lett.*, vol. 20, no. 9, pp. 1760–1763, Sep. 2016.
[34] S. J. Savory, "Congestion aware routing in nonlinear elastic optical networks," *IEEE Photon. Technol. Lett.*, vol. 26, no. 10, pp. 1057–1060, May 15, 2014.
[35] H. V. Stackelberg and A. T. Peacock, "The theory of the market economy," *Economica*, vol. 20, no. 80, p. 384, 1952.
[36] Y.-Y. Tan, Y.-C. Jiao, H. Li, and X.-K. Wang, "MOEA/D + uniform design: A new version of MOEA/D for optimization problems with many objectives," *Comput. & Oper. Res.*, vol. 40, no. 6, pp. 1648–1660, 2013.
[37] V. Eramol, E. Miucci, M. Ammar, and F. G. Lavacca, "An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2008–2025, Aug. 2017.
[38] L. Wang, Z. Lu, X. Wen, R. Knopp, and R. Gupta, "Joint optimization of service function chaining and resource allocation in network function virtualization," *IEEE Access*, vol. 4, pp. 8084–8094, 2017.
[39] X. Wang, Y. Wang, and Y. Cui, "Energy and locality aware load balancing in cloud computing," *Integr. Comput.-Aided Eng.*, vol. 20, no. 4, pp. 361–374, 2013.
[40] Y. Wang, P. Lu, W. Lu, and Z. Zhu, "Cost-efficient virtual network function graph (vNFG) provisioning in multidomain elastic optical networks," *J. Lightw. Technol.*, vol. 35, pp. 2712–2723, Jul. 1, 2017.
[41] M. Xia, M. Shirazipour, Y. Zhang, H. Green, and A. Takacs, "Network function placement for NFV chaining in packet/optical datacenters," *J. Lightw. Technol.*, vol. 33, no. 8, pp. 1565–1570, Apr. 2015.
[42] H. Xuan, S. Wei, Y. Li, and H. Guo, "Off-line time aware scheduling of bag-of-tasks on heterogeneous distributed system," *IEEE Access*, pp. 104777–104788, 2019.
[43] H. Xuan, S. Wei, W. Tong, D. Liu, and C. Qi, "Fault-tolerant scheduling algorithm with re-allocation for divisible task," *IEEE Access*, vol. 6, pp. 73147–73157, 2018.
[44] H. Yang, Q. Yao, A. Yu, Y. Lee, and J. Zhang, "Resource assignment based on dynamic fuzzy clustering in elastic optical networks with multi-core fibers," *IEEE Trans. Commun.*, vol. 67, no. 5, pp. 3457–3469, May 2019.
[45] M. Yannis and M. Magdalene, "A bilevel genetic algorithm for a real life location routing problem," *Int. J. Logistics Res. Appl.*, vol. 11, no. 1, pp. 49–65, 2008.
[46] Z. Ye, X. Cao, J. Wang, H. Yu, and C. Qiao, "Joint topology design and mapping of service function chains for efficient, scalable, and reliable network functions virtualization," *IEEE Netw.*, vol. 30, no. 3, pp. 81–87, May/Jun. 2016.
[47] 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, Jul. 15, 2016.
[48] J. Zhang, J. Luo, and F. Dong, "Scientific workflow scheduling in non-dedicated heterogeneous multicluster with advance reservations," *Integr. Comput.-Aided Eng.*, vol. 22, no. 3, pp. 261–280, 2015.
[49] Y. Zhao, B. Chen, J. Zhang, and X. Wang, "Energy efficiency with sliceable multi-flow transponders and elastic regenerators in survivable virtual optical networks," *IEEE Trans. Commun.*, vol. 64, no. 6, pp. 2539–2550, Jun. 2016.

HEJUN XUAN received the B.Sc. degree in computer science and technology from Xinyang Normal University, China, in 2012, and the Ph.D. degree in computer software and theory from Xidian University, China, in 2018. He is currently with the Computer and Information Technology, Xinyang Normal University. His research interests include cloud/grid/cluster computing, and scheduling in parallel and distributed systems.

SHIWEI WEI received the B.Sc. and M.Sc. degrees in computer science and technology from the Guilin University of Electronic Technology, China, in 2004 and 2007, respectively. He is currently pursuing the Ph.D. degree in computer science and technology with the Xidian University of China. He is also an Assistant Professor with the Guilin University of Aerospace Technology. His research interests include cloud computing and machine learning.

YAN FENG received the Ph.D. degree in applied mathematics from Shantou University, China. He is currently a Professor of computer and information technology with Xinyang Normal University. His research interests include machine learning and data mining.

DAOHUA LIU received the Ph.D. degree in computer science and technology from the Xi'an University of Architecture and Technology, China. He is currently a Professor of computer and information technology with Xinyang Normal University. His research interests include machine learning and image processing.

YANLING LI received the Ph.D. degree in computer science and technology from the Huazhong University of Science and Technology, China. He is currently a Professor of computer and information technology with Xinyang Normal University. His research interests include machine learning and image processing.