Multi-objective Model and Genetic Algorithm for Multisource Multicast VNF Service Chain Deployment Problem

Jintao Liu, Xuelin Zhao, Ruiyan Ma, Hejun Xuan, Xinming Guo, and Wenjie Zhai

***Abstract*—Network function virtualization can increase the versatility of computing and network platforms, enabling them to provide more flexible services. In terms of a software-defined network and network function virtualization environment, this paper explores the joint virtual network function (VNF) deployment and traffic routing in multisource multicast for VNF service chain deployment to minimize the deployment overhead, transmission delay. Firstly, a multi-objective optimization model, which minimizes the deployment overhead and transmission delay, is established to tackle this challenge problem. Then, a highly efficient coding scheme, crossover, and mutation operators are designed. Based on these, an improved genetic algorithm in multi-objective evolutionary algorithm based on decomposition (MOEA/D) framework (GA-MOEA/D) is proposed to solve this multi-objective problem. Finally, simulation experiments are conducted using two widely used network topologies in order to demonstrate the performance of the proposed algorithm. The simulation results demonstrate that the proposed algorithm can obtain the smaller deployment overhead, transmission delay, load degree and energy consumption 5.2%-15.61%, 11.2%-26.4%, 8.3%-17.6% and 7.8%-14.4% than that of the compared algorithm, respectively.**

***Index Terms*—virtual network function, multi-objective, genetic algorithm, multisource multicast**

I. Introduction

In recent years, with the wide application of internet technology, the appearance of network function virtualization and a software-defined network, the management and design of communication networks have undergone fundamental changes [1], [2], [3]. By using NFV technology, the network function software is deployed on a universal server, replacing the intermediate box device in the traditional hardware network, decoupling the network function from the hardware device, thus reducing the deployment cost and making the network scalable and flexible [4]. These kinds of software-based network intermediary boxes, such as a firewall, network address translation, intrusion detection system, or router, are called virtual network functions [5], [6]. With the support of SDN technology, network resources can be allocated on demand and regulated flexibly [7], [8]. The request flows from the source node to the destination node are processed by different network functions in a certain sequence, thus forming a service function chain. Flexibly embedding a VNF service chain has a great impact on the network delay, construction cost, and operation and maintenance overheads [9], [10].

Unicast is the communication between a source and a destination, while multicast is the communication that transmits the same data from the source to a set of destinations [11], [12]. The demand for multicast services in network technology is growing rapidly, such as online video conferencing, media broadcasting, multiplayer games, distributed interactive simulation, and so on [13]. It is estimated that 82% of consumers' internet traffic will be attributed to video streaming [14], which clearly shows the trend shift from point-to-point (unicast) communication to point-to-multipoint and multipoint-to-multipoint communication (multicast). Until now, the VNF deployment and traffic routing of unicast services have been extensively studied [15], and relatively little work has been done on the joint VNF deployment and multicast routing of multicast services [16], [17]. Compared with unicast communication, a multicast communication mode can avoid the repeated transmission of a unicast path, thus significantly reducing the bandwidth utilization in the network.

Embedding a VNF service chain in multicast networks is more challenging than that of a unicast VNF service chain [18]. In multicast services, if only the VNF deployment and routing policies in one-to-one unicast services are used, the deployment and routing paths of the VNF will be repeated, which will lead to high service costs. If the VNF service chain embedding location is close to the source but far from the destination, a large service function tree will be constructed, resulting in a higher link cost [19], [20], [21]. If VNF instances are deployed on multiple NFV nodes, the link configuration cost may be reduced, but the function configuration cost will be increased. Therefore, link costs and VNF deployment costs need to be balanced. Another major challenge is how to embed multiple multicast requests in the same grassroots network with limited node and link resources in order to minimize the total cost. A minimum-cost multicast service function tree (MCMSFT) algorithm

Manuscript received September 11, 2023; revised April 5, 2024. This work was supported by National Natural Science Foundation of Henan Province (No. 232300420424), Science and Technology Department of Henan Province (No.242102211070).

Jintao Liu is a Lecturer at the School of Architecture and Civil Engineering, Xinyang Normal University, Xinyang Henan, 46400, China. (email: jtliu_xynu@126.com)

Xuelin Zhao is a Lecturer at the School of Computer and Information Technology, Xinyang Normal University, Xinyang Henan, 46400, China. (email: xlz_cit@xynu.edu.cn)

Ruiyan Ma is an undergraduate at the School of Computer and Information Technology, Xinyang Normal University, Xinyang Henan, 46400, China. (email: 690454390@qq.com)

Hejun Xuan is an Associate Professor at the School of Computer and Information Technology, Xinyang Normal University, Xinyang Henan, 46400, China. (email: xuanhejun0896@xynu.edu.cn)

Xinming Guo is an undergraduate at the School of Computer and Information Technology, Xinyang Normal University, Xinyang Henan, 46400, China. (email: guoxm2022@163.com)

Wenjie Zhai is an undergraduate at the School of Computer and Information Technology, Xinyang Normal University, Xinyang Henan, 46400, China. (email: zwj18739812069@126.com)

is designed[22], this algorithm proposed a hierarchical path segmentation technology, built a multicast VNF service chain tree, and optimized the request resource allocation problem, but it did not consider the specific order of the VNF service chain. A steiner tree-based approach to minimize the cost of the VNF deployment and routing is proposed [23]. In [24], it studied NFV-enabled multicast technology in SDN to maximize the network throughput. Assuming that the number of servers in each request service chain was no more than a constant, two approximate online algorithms were proposed. In this algorithm, all VNFs were deployed in the same server node, which may cause more link configuration costs. To supported placing a VNF of the multicast tree on scattered network nodes, the optimal service function tree in the multicast task network and proposed a two-stage algorithm (TSA) is designed with the goal of reducing the traffic transmission cost[23]. In [25], Alhussein, et al., considered single-service and multiservice scenarios and studied joint multipath multicast routing and VNF placement with minimal configuration cost and maximum total throughput as the optimization objectives. A multiservice path based multicast service chain VNF-PR model, which allowed partial paths carrying the same data to be merged into one path even if the service requests were different, is establised[26]. The optimal embedding problem of an NFV-enabled multicast service chain is described as a mixed binary linear program in[27], and then adopted a continuous upper bound minimization algorithm based on punishment to solve this problem in a reasonable time.

As for the research on multisource multicast, the total cost of all allocated VNFs and all multicast trees in the forest is minimized to cover the forest with services[28]. An approximate algorithm was designed to solve this problem, but this method only considered placing one VNF for each NFV node. In [29], this paper designed a heuristic algorithm for the construction of a multisource multicast tree. Although the limitation of bandwidth and delay was considered, the algorithm aimed to find a common link for all destination users and embed the service function chain. However, in a large network, due to the link cost between destinations, this method could not guarantee the optimal cost. A NFV multisource multicast resource optimization model is proposed in [30]. The decomposition model was used to decompose it into a main problem and several pricing problems to optimize the problem, but the optimization goal was only to maximize the request acceptance rate, and the cost optimization problem was not taken into account. To summarize, most of the current research work has aimed to restrict all destinations to share the same VNF service chain, which resulted in low efficiency and a high link cost when the destinations were geographically dispersed. Routing costs are increased if all types of network function instances are deployed in one NFV node. An effective solution is to replicate or deploy function instances in multiple NFV nodes to reduce the link costs due to flexible routing. In the NFV of multicast, there is little research on multisource multicast (multipoint-to-multipoint) communication. The objective of this paper is to embed a VNF service chain in the multicast network and minimize the overall node and link resource consumption cost under the constraint of the multicast request delay. How to reduce network resource consumption and load pressure while ensuring the corresponding services for users. an NFV architecture for SFC deployment is introduced [31], and illustrate the VNF service chain orchestration process which is based on SRv6 in multi-domain scenario. Then, we propose an effective SFC dynamic orchestration algorithm. A practical byzantine fault-tolerant consensus mechanism based on reputation value is proposed to save consensus cost and improve consensus efficiency. Combined with AI technology, the in-depth reinforcement learning is introduced into the business function chain choreography, and the business function chain choreography algorithm based on asynchronous advantage actor-critic is designed to optimize the choreography cost [32]. The multi-criteria heuristic (MCH) is used to arrange and route the online VNF. In the offline process, the meta-heuristics based on genetic algorithm (GA) is used to learn the super parameters of the online MCH model to minimize the total power consumption of the NFV infrastructure. The optimization process using genetic algorithm is only performed once before the main operation of NFV [33].

This paper studies the optimization of multisource multicast resources for multiple service requests. Firstly, in view of the VNF deployment rules, transmission resource constraints, and timely delay, joint multicast VNF deployment and routing problems are described and defined mathematically, and a multi-objective optimization model is established. Then, a highly efficient coding scheme, crossing, and mutation operators are designed. Based on these, an improved genetic algorithm in MOEA/D framework is proposed to solve this multi-objective model. Finally, simulation experiments are conducted in two widely used network topologies in order to demonstrate the performance of the proposed algorithm.

Movitation: In this paper, we propose a new solution for multisource multicast VNF service chain deployment problem. There are more and more researches focus on multisource multicast VNF service chain deployment problem. In them, most of them established an integer linear programming model to minimize a single goal. However, network service providers may not only optimize a certain goal, but optimize multiple goals, so that multiple indicators of the network can reach the optimal when making decisions. So, single global optimization model is not appropriate. Thus, we establish a multi-objective optimization model for the multisource multicast VNF service chain deployment problem to meet multiple needs of decision makers. Moreover, how to solve the optimization model efficiently is also a hard work. Thus, we propose a multi-objective genetic algorithm to solve the established model.

Contributions: In summary, the innovation of our works are summarized as follows:

- A detailed analysis on multisource multicast VNF service chain deployment problem is given. To determine theoptimal VNF service chain strategy, a multi-objective optimization model for the multisource multicast VNF service chain deployment problem is established.
- A highly efficient coding scheme, crossing, and mutation operators are designed. Based on these, an improved genetic algorithm in MOEA/D framework is proposed to solve this multi-objective model.
- For the sake of demonstrating the performance of the proposed algorithm, simulation experiments are conducted by using two widely used network topologies.

II. Network Destination and Mathematical Modeling

A. Network Model

1) Physical Network: In this paper, a software-defined physical base network is represented as an undirected graph $G = (V, L)$, where V indicates the set of network nodes. $V = V_s \bigcup V_d \bigcup V_m \bigcup V_w$, where V_s represents the set of source nodes, used to send traffic. V_d indicates the destination node set. V_m represents a set of nodes connected to the computing servers. These server nodes serve as NFV nodes and can deploy various network functions. The total capacity of the computing resources on node u is expressed as C_u. V_w represents the set of common switch nodes. The switch is only used for copying or forwarding traffic, so its node capacity is 0. The other parameter symbols in the physical network are shown in Table I.

TABLE I
THE PARAMETER SYMBOLS IN THE PHYSICAL NETWORK.

Symbols	Description
V	The set of nodes
L	The set of links
V_s	The set of source nodes
V_d	The set of destination nodes
V_m	The set of nodes connected to the computing servers
V_w	The set of common switch nodes
N_u	The set of adjacent nodes of node u
$l_{u,v}$	The link from node u to node v, $\forall l_{u,v} \in L$
C_u	The total capacity of the computing resources on node u
$B_{u,v}$	The total bandwidth on link $l_{u,v}$
$\tau_{u,v}$	The time delay of the link l_u, v

2) VNF Service Chain: We assume an all VNF F type collection, where $\forall f_j \in F$ is defined as a specific VNF, we choose a few f_j from an F connection in a particular order, and the VNF service chain can form a service chain, namely, the VNF service chain $f_{c_k} = (f_1, f_2, \cdots, f_{|k|})$. Fig. 1 shows a multicast service chain sample, where data packets from the source node arrive at the destination node in turn after the service chain. In this multicast VNF service chain, it has two destination mode, Node 1 and node 2. Three VNF (VNF1, VNF2 and VNF3) should be deployment. In addition, that three VNF has a special queue. It is assumed that each V_m supports all types of VNFs in F and can run multiple VNFs. The k-th multicast VNF service chain $r_k \in R$ is represented as $r_k = \{s_k, d_k, f_{c_k}, t_k\}$, and the other parameter symbols are shown in Table I.

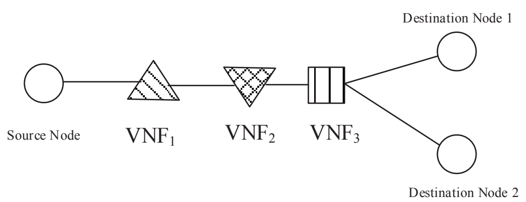


Fig. 1. The multicast VNF service chain.

3) Description of the Variables: In the k-th multicast VNF service chain r_k, if the data flow to the destination node, $d \in d_k$ has passed the link $l_{u,v}$ in the path from VNF f_j to VNF f_{j+1}, and $\varepsilon^{k,d}_{f_j,u,v} = 1$; otherwise, $\varepsilon^{k,d}_{f_j,u,v} = 0$. For the k-th multicast VNF service chain r_k, $\chi^k_{f_j,u,v} = 1$ denotes that the link $l_{u,v}$ is on the path from VNF f_j to VNF f_{j+1} in the multicast service function; otherwise, $\chi^k_{f_j,u,v} = 0$. For the k-th multicast VNF service chain r_k, the destination node $d \in d_k$, VNF instance f_j is newly deployed on server node u, and $\theta^{k,d}_{f_j,u}$=1; otherwise, it is 0. Similarly, if VNF f_j in the VNF service chain set R is newly deployed on NFV node u, $\lambda_{f_j,u} = 1$, otherwise, it is 0.

TABLE II
THE PARAMETER SYMBOLS IN THE VNF SERVICE CHAIN.

Symbols	Description
F	The set of all types of VNFs
R	The set of all multicast VNF service chains
r_k	The k-th multicast VNF service chain
f_{c_k}	The VNF in r_k
$f_{c_{k'}}$	The VNF is added to the source and destination of r_k
s_k	The source node of the r_k, $\forall s_k \in V_s$
d_k	The destination node of the r_k, $\forall d_k \in V_d$
δ_{f_k}	The computing resources required when a new f_k is deployed
$\delta_{f_k,u}$	The deployment cost required to deploy f_k on node u
t_k	The time delay requirement of r_k
b_k	The traffic size of r_k
σ_k	The delay between the destination of r_k

B. Multi-objective Optimization Model

Aiming at the optimization of multisource multicast resources for multiple service requests, this paper designs a multisource multicast multi-objective optimization model, which comprehensively considers the node resources, the link resources, the routing traffic relationship, and the multicast delay requirements in the network and finds the optimal embedding position of the service function chain. The first goal is to minimize the deployment overhead in the network, and this objective is described as:

$$\min \left\{ \sum_{u \in V_m} \sum_{f_j \in F} \lambda_{f_j,u} \delta_{f_j,u} \right\}. \tag{1}$$

Since $\lambda_{f_j,u} \in \{0, 1\}$, we have $\sum_{u \in V_m} \sum_{f_j \in F} \lambda_{f_j,u} \delta_{f_j,u} \leq \sum_{u \in V_m} \sum_{f_j \in F} \delta_{f_j,u}$. We can normalize this objective as

$$\min f_1 = \min \left\{ \frac{\sum\limits_{u \in V_m} \sum\limits_{f_j \in F} \lambda_{f_j,u} \delta_{f_j,u}}{\sum\limits_{u \in V_m} \sum\limits_{f_j \in F} \delta_{f_j,u}} \right\}. \tag{2}$$

Thus, we have $0 \leq f_1 \leq 1$. Another objective is to minimize the transmission delay, and this objective is described as:

$$\min \left\{ \sum_{r_k \in R} \sum_{u,v \in V} \sum_{d \in d_k} \sum_{f_j \in f_{c_k}} \left(\varepsilon^{k,d}_{f_j,u,v} b_k \right) \right\}. \tag{3}$$

Similarly, we can also normalize this objective as

$$\min f_2 = \min \left\{ \frac{\sum\limits_{r_k \in R} \sum\limits_{u,v \in V} \sum\limits_{d \in d_k} \sum\limits_{f_j \in f_{c_k}} \varepsilon^{k,d}_{f_j,u,v} b_k}{\sum\limits_{r_k \in R} \sum\limits_{u,v \in V} \sum\limits_{d \in d_k} \sum\limits_{f_j \in f_{c_k}} b_k} \right\}. \tag{4}$$

From the above analysis, $0 \leq f_1 \leq 1$. To achieve these two goals, some conditions should be satisfied as follows: (a) For each NFV node, the available resources allocated to the VNF should not exceed its node capacity; that is,

$$\sum_{f_j \in F} \lambda_{f_j,u} \delta_{f_j} \leq C_u \forall u \in V. \tag{5}$$

(b) The resource usage of each link should not exceed its link capacity,

$$\sum_{r_k \in R} \sum_{u,v \in V} \sum_{f_j \in f_{c_k}} \chi^k_{f_j,u,v} b_k \leq C_{u,v}, \forall u,v \in V. \quad (6)$$

(c) If f_0 is used as the network function of the source node in $f^{'}_{c_k}$, other nodes except the source node cannot obtain the service of f_0. For each destination node d, only one source node among multiple source nodes can obtain the service of f_0,

$$\sum_{u \in s_k} \theta^{k,d}_{f_0,u} = 1, \forall r_k \in R, d \in d_k. \quad (7)$$

(d) For each VNF f_j, it only can be deployed on one NFV node and cannot be shared by multiple NFV nodes,

$$\sum_{u \in V_m} \theta^{k,d}_{f_j,u} = 1, \forall r_k \in R, f_j \in f^{'}_{c_k}, d \in d_k. \quad (8)$$

(e) The multicast VNF service chain on the network must comply with traffic conservation. That is, the outgoing traffic of each node is equal to the incoming traffic, and the multicast VNF service chain requirements must be ensured,

$$\sum_{v \in N_u} \varepsilon^{k,d}_{f_j,u,v} - \sum_{v \in N_u} \varepsilon^{k,d}_{f_j,v,u} = \theta^{k,d}_{f_j,u} - \theta^{k,d}_{f_{j+1},u}, \forall r_k \in R, f_j \in f^{'}_{c_k}. \quad (9)$$

(f) The traffic of the multicast VNF service chain also needs to meet constraints. For destination node $d \in d_k$, if $\varepsilon^{k,d}_{f_j,u,v} = 1$, then $\chi^k_{f_j,u,v} = 1$,

$$\varepsilon^{k,d}_{f_j,u,v} \leq \chi^k_{f_j,u,v}, \forall r_k \in R, f_j \in f^{'}_{c_k}, d \in d_k. \quad (10)$$

(g) We need to ensure that each multicast VNF service chain meets the end-to-end delay constraints,

$$\sum_{u,v \in V} \varepsilon^{k,d}_{f_j,u,v} \tau_{u,v} \leq t_k, \forall d \in d_k, r_k \in R. \quad (11)$$

(h) The delay between all destinations in each multicast VNF service chain should not exceed π_k,

$$\left| \sum_{u,v \in V} \varepsilon^{k,d_m}_{f_j,u,v} \tau_{u,v} - \sum_{u,v \in V} \varepsilon^{k,d_n}_{f_j,u,v} \tau_{u,v} \right| \leq \pi_k, \forall d_m, d_n \in d_k. \quad (12)$$

III. Proposed Genetic Algorithm

A. Algorithm Description

In this paper, a $0-1$ matrix coding method was adopted, which treated the whole matrix as the individual genetic offspring, to ensure the integrity of the individual genes of the offspring. Based on this, the matrix was taken as an individual for genetic calculation. The number of rows in the matrix (represented by m) was equal to the sum of the number of VNFs required in the multicast VNF service chain to be migrated, and the number of columns in the matrix (represented by n) was equal to the number of nodes in the underlying physical network. The number of individuals in the k generation population of the offspring was N, and each individual was also called a matrix chromosome, which is a matrix of order $m \times n$. Then, $Q_k = \{A_1, A_2, \cdots, A_N\}$ represented the population, where $A^r_k = (a_{ij})_{m \times n}$ represented the r-th individual in the k-generation population, and each element in the individual a^r_{ij} was the gene element of matrix chromosome. a^r_{ij} needed to satisfy the following two constraints, $\sum_{j=1}^{n} a^r_{ij} = 1, i \in \{1, 2, \cdots, m\}$ and $a^r_{ij} \in \{0, 1\}, i \in \{1, 2, \cdots, m\}, j \in \{1, 2, \cdots, n\}$, and the relevant operations on the chromosomes are described as follows:

(1) Population initialization. In this algorithm, the initial population contained N chromosomes, where N was constant and dependent on the size of the problem. In this paper, uniform design is adopted to generate uniformly distributed starting search points. The starting search points can be evenly distributed in the solution space to improve the search ability.

(2) Selection. The elite retention strategy based on the traditional roulette choice method was used. The elite retention strategy can make the population converge to the optimal solution of the solved optimization problem.

(3) Crossover. Multiline matrix hybridization was adopted. According to the hybridization probability $P_c (0 \leq P_c \leq 1)$, the line gene elements in the corresponding positions of two matrix chromosomes were exchanged. The exchanged lines were random. For the crossover operator of two chromosomes A^i_k and A^j_k in population Qk, the initialization was carried out first: $A^i_k = (a^i_{ij})_{mn}$, $A^j_k = (a^j_{i,j})_{mn}$. If the second row exchange is taken as an example, then the matrix chromosome after the crossover operation was:

$$A^i_k = \begin{pmatrix} a^i_{11} & a^i_{12}2 & \cdots & a^i_{1n} \\ a^i_{21} & a^i_{22}2 & \cdots & a^i_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a^i_{m1} & a^i_{m2}2 & \cdots & a^i_{mn} \end{pmatrix}$$

$$A^j_k = \begin{pmatrix} a^j_{11} & a^j_{12} & \cdots & a^j_{1n} \\ a^j_{21} & a^j_{22} & \cdots & a^j_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a^j_{m1} & a^j_{m2} & \cdots & a^j_{mn} \end{pmatrix}.$$

(4) Mutation. This paper adopted the adaptive value according to the fitness change, and the formula was as follows:

$$P^i_m = \begin{cases} \frac{fit_{max} - fit(A^i_k)}{fit_{max} - fit_{min}}, & fit(A^i_k) \leq \overline{fit} \\ P_m, & fit(A^i_k) > \overline{fit} \end{cases}. \quad (13)$$

This calculation of mutation probability ensured that individuals with a low fitness had a greater probability of mutation, thus increasing the possibility of transforming individuals into individuals with high fitness. For matrix-encoded chromosomes with constraints, the mutation operation was carried out on each row. In the mutation operation, gene elements of a certain row or some rows in a matrix chromosome were changed according to probability P^i_m, and the choice of the changed rows was random, but the condition had to meet the constraints of the problem itself.

(5) Selection. Selecting the right individuals to enter the next generation is conducive to obtaining the frontier with good convergence, diversity and uniformity, and providing better choices for decision makers. This article uses the following steps to select individuals. This chapter defines a measure of the merits and demerits. The calculation method is as follows:

$$\psi_i = \alpha_{gen}(1 - \chi_i) + (1 - \alpha_{gen})\gamma_i, \quad (14)$$

where χ_i and γ_i the constraint violation degree and domination degree of the i-th solution respectively. The degree of constraint violation is the proportion of constraint violation of the solution. The smaller the degree of constraint violation, the better. When the degree of constraint violation is 0, the solution is a feasible solution. Dominance refers to the proportion of the number of other solutions dominated by this solution in all solutions. The higher the proportion, the higher the dominance degree and the better the solution. α_{gen} represents the weight of regulation constraint violation degree and dominance degree, which is calculated as follows: After the operations of crossover and mutation, many new individuals were obtained, but these individuals may not be feasible solutions to the original problem. Therefore, it was necessary to add feasibility tests after genetic manipulation to ensure that each newly generated individual was within the range of the feasible solutions to the original problem. Each chromosome corresponded to the VNFs' deployment and instantiation, respectively, from which the value of the target variable $x_{i,r}^u$ could be determined. At this point, we judged whether the constraint of node deployment in the model was established. Under the condition that the constraint was established, the value of another target variable α was obtained according to the VNFs' mapping scheme combined with the path selection algorithm. In this way, we could judge whether the constraints of the link mapping in the model were satisfied. When these constraints were satisfied, we obtained the feasible solution of the problem.

$$\alpha = \begin{cases} 1, & \mu \leq 0.5 \\ 1 - \frac{g}{G_{max}}, & else \end{cases}, \tag{15}$$

where g represents the current algebra, G_{max} represents the maximum algebra of the algorithm iteration, and B represents the proportion of feasible solutions in all solutions, that is, when the proportion of feasible solutions is less than half, the metric index of feasible solutions is 1. This formula shows that in the initial iteration, more attention is paid to the degree of constraint violation, that is, the feasible solution is selected as far as possible. As the iteration progresses, more and more feasible solutions are generated, so the non-dominant solution should be selected and the solution with higher dominance should be selected.

B. Fitness Calculation

If the fitness function corresponding to a chromosome has a large value, it indicates that the chromosome is close to the optimal solution, and it is more likely to be selected to generate the next generation population. In this case, the average fitness of the population and the optimal fitness of the individual can be improved generation by generation to solve the optimization problem. Each network function deployment relationship can be encoded to obtain the corresponding chromosome individual. Conversely, a network function deployment can be obtained by decoding each individual chromosome. Therefore, for a given chromosome, the original problem can be reduced to a link mapping problem. Through the path selection of the problem, the fitness of the chromosome and the corresponding optimal link mapping scheme can be obtained. For the minimization problem in this paper, the following fitness function was adopted:

$$fit_i(x) = 1 - f_i(x), \tag{16}$$

where $f_i(x)$ was the i-th optimization objective function in the system model. Since we had $0 \leq f_i(x) \leq 1$, and both objectives of the multi-objective optimization model were minimization problems, we defined the objective as $fit_i(x) = 1 - f_i(x)$. The greater the fitness value, the better the individual.

IV. Experimental Results and Analysis

A. Parameter Setting

This experiment was run on a Windows PC with Intel(R) Core(TM) i7 6400-CPU 3.4Ghz and 128Gb memory. Two widely used network topographies are used, i.e., CHNNET and ARPANET, respectively. CHNNET is China's public Internet network based on Internet network technology operated and managed by China Telecom, and is the backbone of China's Internet network. It has 15 nodes and 27 links. ARPANET belongs to the second generation network, which is a computer network centered on communication subnet. It is a new network resource exchange mode after the first generation network. It includes 20 nodes and 32 links. The effects of the number of multicast requests, the number of destination nodes, the length of the VNF service chain (the number of VNFs), and the number of source nodes on the total cost, link utilization, and total delay were evaluated. Under the same parameters, the average results of all indexes were obtained through 10 rounds of training. The other parameters were set as follows: we assumed that the total resource capacity of each node was 3000Mbps, the link capacity was set as the random distribution of [1000, 2000] Mbps, the unit cost of a link and node resources was 1, and the resource required for deploying one VNF was 100Mbps. We assumed that each server node in the network service supported multiple instances of VNF and supported all types of VNFs. Set F contained seven types of VNF, each of which had a deployment cost of $\{10, 20, 30, 40, 30, 20, 10\}$. The sequence of each VNF service chain and the number of VNFs were randomly generated, and all source and destination nodes were randomly generated. The size of each stream followed the normal distribution with the mean value of 20 and the standard deviation of 4. The time delay of each edge was randomly selected from [15, 50]. The total time delay demand of each multicast request met the normal distribution with the mean value of 400 and the standard deviation of 8. Population size and maximum number of iterations are select as 100 and $10000 \times N_{VNF-SC}$, where N_{VNF-SC} denotes the number of VNF service chains. Crossover and mutation probability are selected 0.75 and 0.08, respectively.

B. Experimental Results and Analysis

There is no revealed algorithm to tackle the problem of multi-objective model for multisource multicast VNF service chain deployment. Thus, some relevant algorithm should be selected to show the performance of the proposed algorithm. Three benchmark algorithms have been compared with proposed algorithm to show the performance of the proposed algorithm. We chose a relatively simple network topology and simulated the performance of the FFD [34], CCFGE [35], and ILP-PSO [36] algorithms.

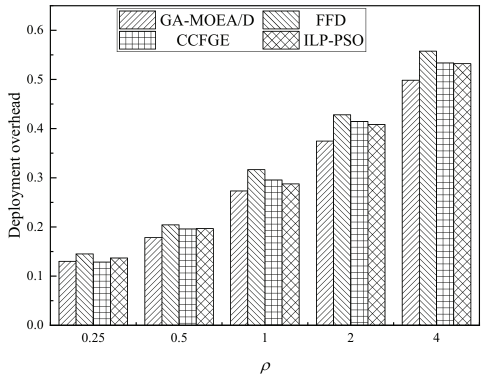
The first, FFD is an overlay network architecture to augment enhanced multimedia broadcast multimedia service (eMBMS) to address the limitations of the standard eMBMS architecture and enable service-less multicast for crowd source live video providers. A virtual network function (VNF) service that identifies potential multicast scenarios based on user requests for a live video within a confined area. The VNF Application Server collects information, validates a potential multicast scenario, and initiates an ad-hoc multicast service on the fly. The second, CCFGE considers the problem of provisioning multi-source multicast services where each service consists of a set of in-network virtual functions that must be chained in a particular order to meet the quality of service demanded by end users. It deal with a reliable service where reliability is attained by provisioning backup functions for the service. The third, the integer encoding scheme is problem-specific and offers a natural way to represent VNF-P solutions, and the proposed wolf position update mechanism divides the wolf pack into two groups in each iteration, where one group performs exploitation while the other focuses on global exploration. It provides the search with a balanced local exploitation and global exploration during evolution.

To demonstrate the performance of the proposed algorithm, four indicators are used. Since the two objectives are deployment overhead and transmission delay, so deployment overhead and transmission delay are selected as two indicators. In addition, energy consumption and load balance are two very important indicators of the network parameters. Thus, energy consumption and load balance are both used as the indicators to show the performance of the proposed algorithm.

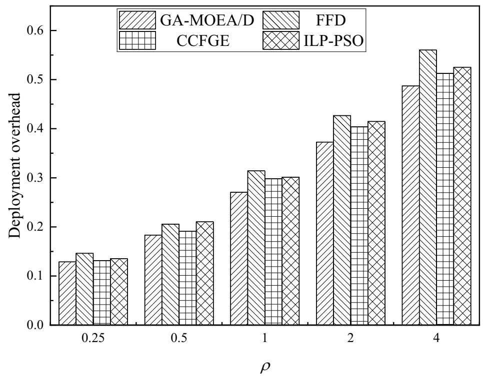
The number of data center nodes was fixed $N_D = N_V/4$, $N_D = 2N_V/$ and $N_D = 3N_V/4$. In each experiment, the number of VNF-SCs was set as $N_R = \rho N_V(N_V - 1)$, and $\rho = 0.25$, 0.5, 1, 2, and 4, respectively. Figures 2 through 4 show $N_D = N_V/4$, $N_D = 2N_V/$ and $N_D = 3N_V/4$.

Fig. 2 to Fig. 4 show the deployment overhead obtained by GA-MOEA/D algorithm proposed in this chapter and three comparison algorithms in the two networks. Fig. 2 shows the deployment overhead between the two networks when $N_D = N_V/4$. It can be seen from the experimental results that under the condition of the same number of VNF service chains, the deployment overhead by GA-MOEA/D algorithm is smaller than that obtained by the comparison algorithm. Similarly, when $N_D = N_V/4$ and $N_D = 3N_V/4$, the deployment overhead obtained is shown in Fig. 3 and Fig. 4 respectively. It can be seen from the experimental results that GA-MOEA/D achieves a lower average delay than the comparison algorithm under the same number of VNF service chains. In each simulation diagram, the deployment overhead increases with the increase of the number of VNF service chains. When $N_D = N_V/2$ and the number of VNF service chains is $N_R = 0.25N_V(N_V - 1)$, the average delay obtained by GA-MOEA/D is 3.5% less than the deployment overhead obtained by the comparison algorithm. When the number of VNF service chains is $N_R = 4N_V(N_V - 1)$, the average delay obtained by GA-MOEA/D is 8.5% less than that obtained by the comparison algorithm.

Fig. 5 to Fig. 7 show the average time delay obtained by GA-MOEA/D algorithm proposed in this chapter and three comparison algorithms in the two networks. Fig. 5

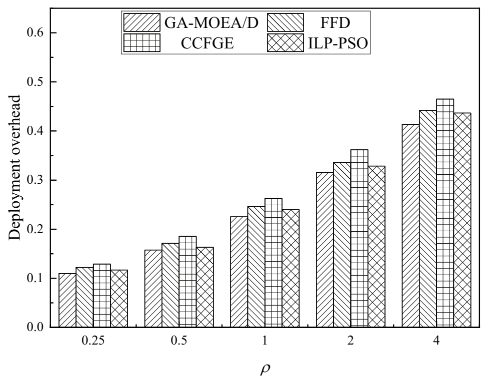


(a) Deployment overhead obtained in CHNNET.

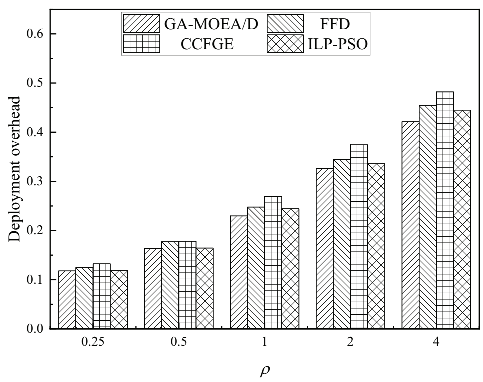


(b) Deployment overhead obtained in ARPANET.

Fig. 2. Deployment overhead obtained when $N_D = N_V/4$.

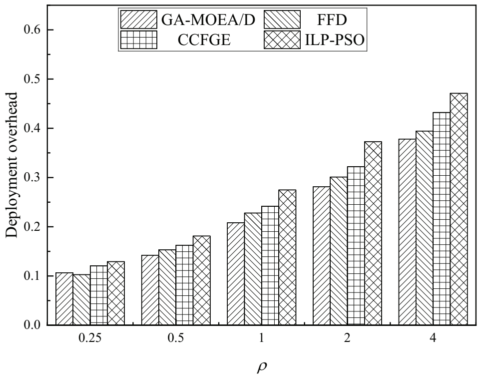


(a) Deployment overhead obtained in CHNNET.

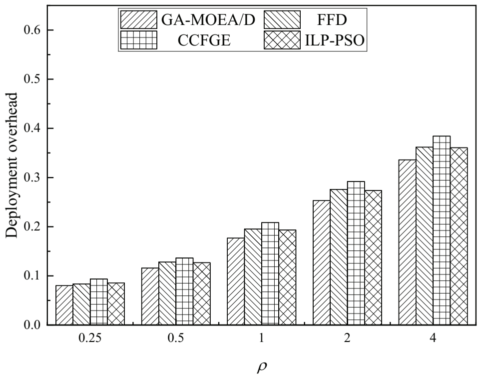


(b) Deployment overhead obtained in ARPANET.

Fig. 3. Deployment overhead obtained when $N_D = N_V/2$.

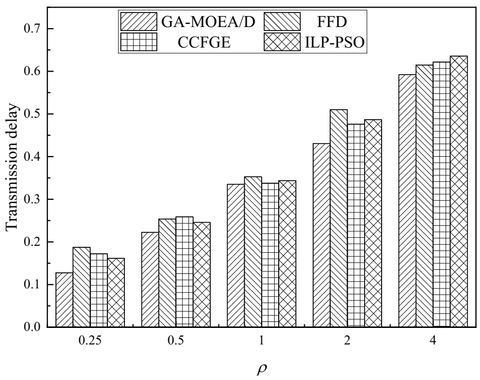


(a) Deployment overhead obtained in CHNNET.

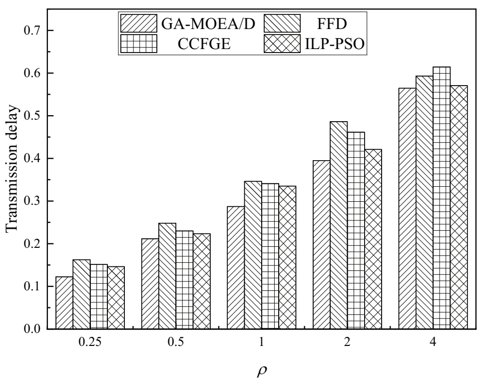


(b) Deployment overhead obtained in ARPANET.

Fig. 4. Deployment overhead obtained when $N_D = 3N_V/4$.

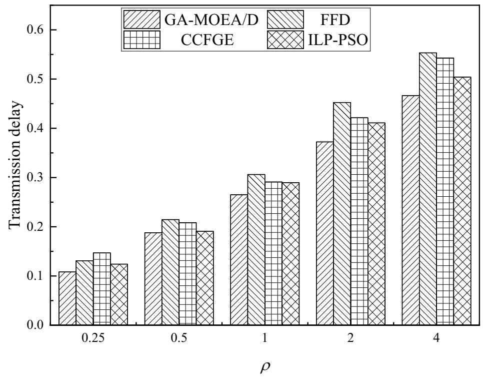


(a) Transmission delay obtained in CHNNET.

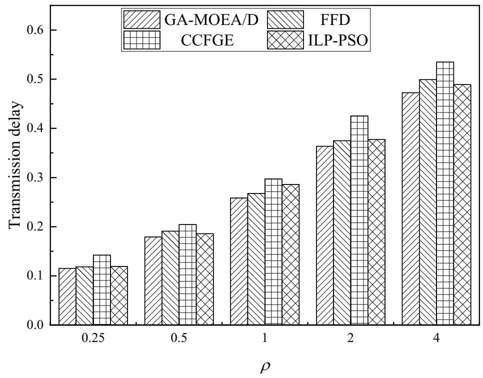


(b) Transmission delay obtained in ARPANET.

Fig. 5. Transmission delay obtained when $N_D = N_V/4$.

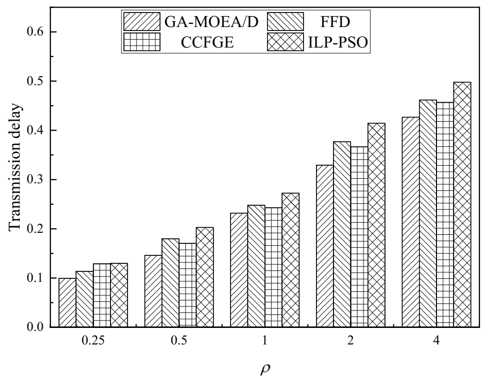


(a) Transmission delay obtained in CHNNET.

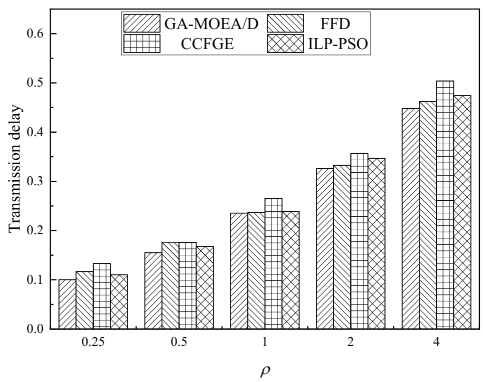


(b) Transmission delay obtained in ARPANET.

Fig. 6. Transmission delay obtained when $N_D = N_V/2$.

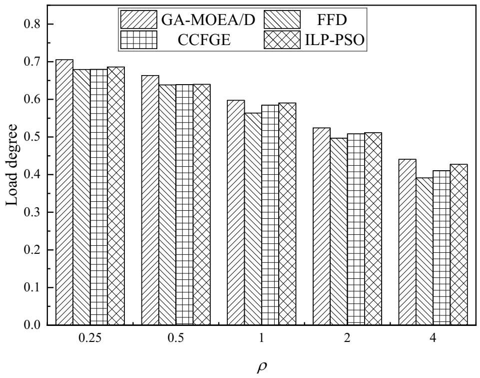


(a) Transmission delay obtained in CHNNET.

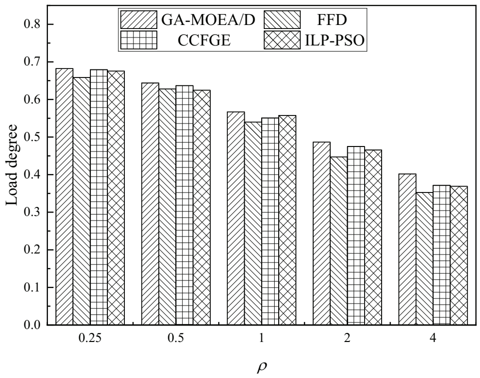


(b) Transmission delay obtained in ARPANET.

Fig. 7. Transmission delay obtained when $N_D = 3N_V/4$.

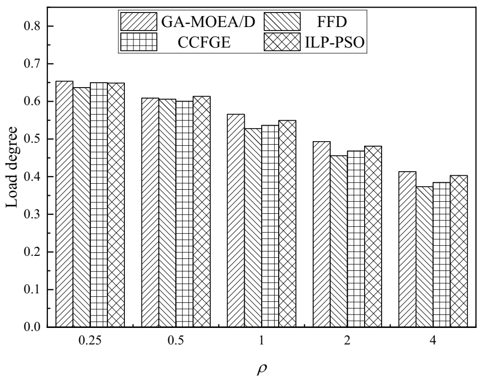


(a) Load degree obtained in CHNNET.

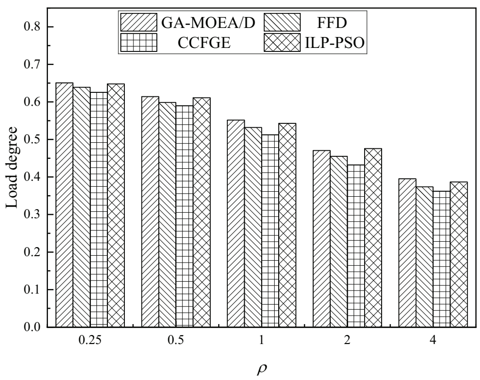


(b) Load degree obtained in ARPANET.

Fig. 8. Load degree obtained when $N_D = N_V/4$.

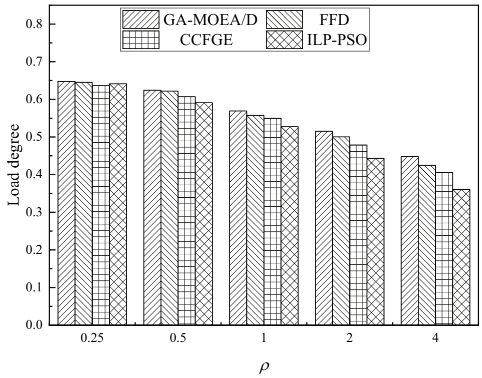


(a) Load degree obtained in CHNNET.

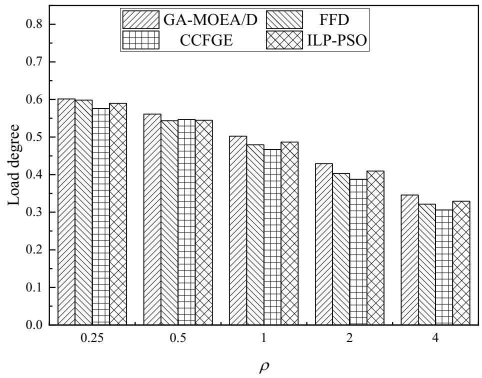


(b) Load degree obtained in ARPANET.

Fig. 9. Load degree obtained when $N_D = N_V/2$.

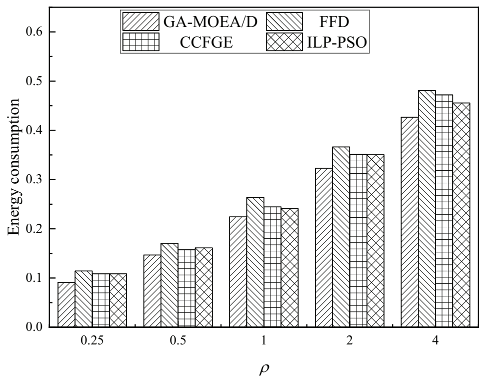


(a) Load degree obtained in CHNNET.

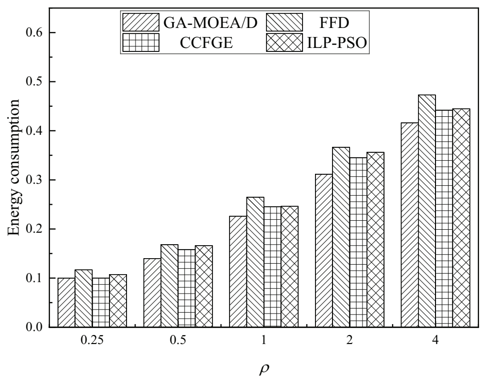


(b) Load degree obtained in ARPANET.

Fig. 10. Load degree obtained when $N_D = 3N_V/4$.

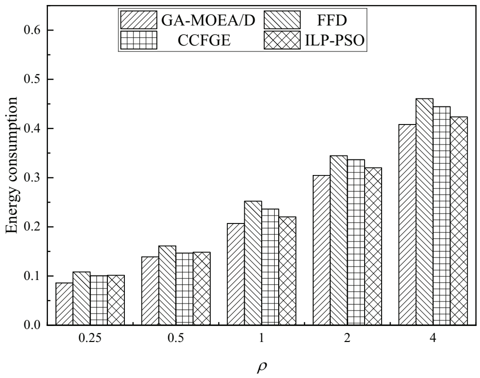


(a) Energy consumption obtained in CHNNET.

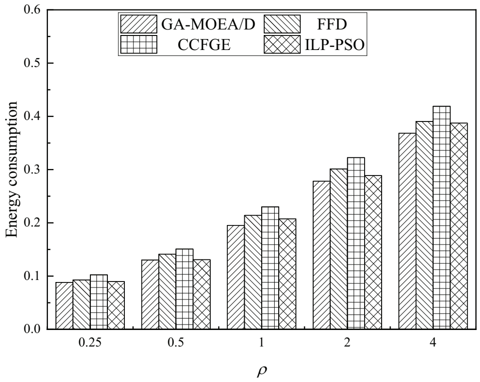


(b) Energy consumption obtained in ARPANET.

Fig. 11. Energy consumption obtained when $N_D = N_V/4$.

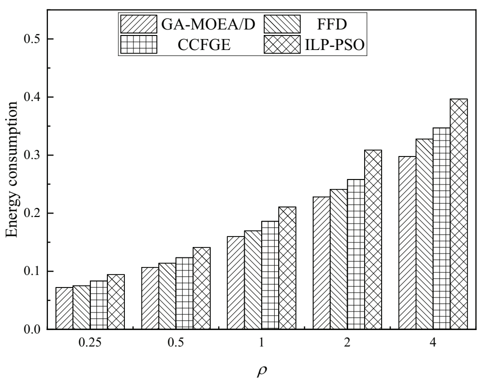


(a) Energy consumption obtained in CHNNET.

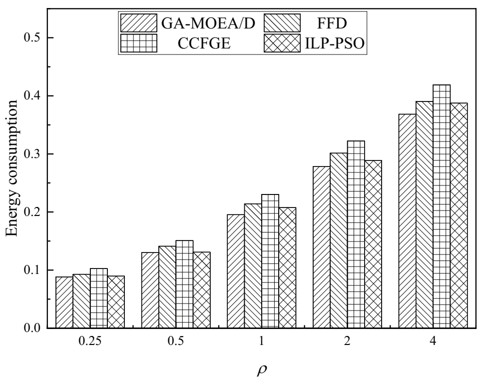


(b) Energy consumption obtained in ARPANET.

Fig. 12. Energy consumption obtained when $N_D = N_V/2$.



(a) Energy consumption obtained in CHNNET.



(b) Energy consumption obtained in ARPANET.

Fig. 13. Energy consumption obtained when $N_D = 3N_V/4$.

shows the average latency between the two networks when $N_D = N_V/4$. It can be seen from the experimental results that, under the condition of the same number of VNF service chains, the average delay obtained by GA-MOEA/D algorithm is smaller than that obtained by the comparison algorithm. Similarly, when $N_D = N_V/4$ and $N_D = 3N_V/4$, the average delay obtained is shown in Fig. 6 and Fig. 7 respectively. It can be seen from the experimental results that GA-MOEA/D achieves a lower average delay than the comparison algorithm under the same number of VNF service chains. In each simulation diagram, the average delay increases with the increase of the number of VNF service chains. When $N_D = N_V/2$ and the number of VNF service chains is $N_R = 0.25N_V(N_V - 1)$, the average delay obtained by GA-MOEA/D is 3.5% less than the total delay obtained by the comparison algorithm. When the number of VNF service chains is $N_R = 4N_V(N_V - 1)$, the average delay obtained by GA-MOEA/D is 8.5% less than that obtained by the comparison algorithm.

Fig. 8 to Fig. 10 show the experimental results of load degree obtained by GA-MOEA/D, the algorithm proposed in this chapter, and three comparison algorithms in two networks. When $N_D = N_V/4$ is shown in Fig. 8, load balancing degree is obtained in two networks. It can be seen from the experimental results that the load balancing degree by GA-MOEA/D under the same number of VNF service chains is greater than that of the three comparison algorithms. Similarly, when $N_D = N_V/2$ and $N_D = 3N_V/4$, the load balancing degree obtained is shown in Fig. 9 and Fig. 10. It can be seen from the experimental results that GA-MOEA/D gets greater load balancing degree than the comparison algorithm when the number of functional service chains of virtual network is the same. In each simulation diagram, the load balancing degree increases with the decrease of the number of VNF service chains. When $N_D = N_V/2$, the number of VNF service chains is $N_R = 0.25N_V(N_V - 1)$, the load balancing degree obtained by GA-MOEA/D is 4.6% less than that obtained by the comparison algorithm. When the number of VNF service chains is $N_R = 4N_V(N_V - 1)$, the load balancing degree obtained by GA-MOEA/D is 8.9% less than that obtained by the comparison algorithm.

Fig. 11 to Fig. 13 show the experimental results of energy consumption obtained by GA-MOEA/D, the algorithm proposed in this chapter, and three comparison algorithms in two networks. When $N_D = N_V/4$ is shown in Fig. 11, energy consumption is obtained in two networks. It can be seen from the experimental results that the energy consumption obtained by GA-MOEA/D under the same number of VNF service chains is less than that of the three comparison algorithms. Similarly, when $N_D = N_V/2$ and $N_D = 3N_V/4$, the energy consumption obtained is shown in Fig. 12 and Fig. 13. It can be seen from the experimental results that GA-MOEA/D gets less energy consumption than the comparison algorithm when the number of functional service chains of virtual network is the same. In each simulation diagram, the energy consumption increases with the increase of the number of VNF service chains. When $N_D = N_V/2$, the number of VNF service chains is $N_R = 0.25N_V(N_V - 1)$, the energy consumption obtained by GA-MOEA/D is 3.5% less than that obtained by the comparison algorithm. When the number of VNF service chains is $N_R = 4N_V(N_V - 1)$, the energy consumption obtained by GA-MOEA/D is 9.5% less than that obtained by the comparison algorithm.

V. Conclusion and Future Work

In this paper, a multi-objective optimization model was established. It can provide decision-makers with more decision schemes and a new way of thinking and modeling for studying other problems, such as taxonomy of controller placement problem, Self-organized design of virtual reality simulator and so on. A high efficient algorithm based on MOEA/D was proposed. It promotes the development of intelligent algorithms and opens up new application scenarios and fields.

In future research, we will intend to deepen our research in the following aspects. First, a further improve the search ability, high efficiency, scalability of GA-MOEA/D should be considered so that it can simultaneously adapt to different types of network topologies. In addition, how to successfully apply the studied algorithm to the actual network is also the focus of future research.

References

[1] Y. He, X. Zhang, Z. Xia, Y. Liu, K. Sood, and S. Yu, "Joint optimization of service chain graph design and mapping in NFV-enabled networks," *Computer Networks*, vol. 202, p. 108626, 2022.

[2] P. Dhavakumar and N. P. Gopalan, "An efficient parameter optimization of software reliability growth model by using chaotic grey wolf optimization algorithm," *Journal of Ambient Intelligence and Humanized Computing*, no. 12, pp. 3177–3188, 2021.

[3] V. Eramo and T. Catena, "Application of an innovative convolutional/LSTM neural network for computing resource allocation in nfv network architectures," *IEEE Transactions on Network and Service Management*, vol. 19, pp. 2929–2943, 2022.

[4] S. Kohei, "Research challenges for network function virtualization re-architecting middlebox for high performance and efficient, elastic and resilient platform to create new services," *IEICE Transactions on Communications*, vol. 101, no. 1, pp. 96–122, 2018.

[5] J. Li, W. Liang, Z. Xu, X. Jia, and W. Zhou, "Service provisioning for multi-source IoT applications in mobile edge computing," *ACM Transactions on Sensor Networks*, vol. 18, no. 2, pp. 1–25, 2021.

[6] S. Y. Li, K. B. Yuan, Y. Zhang, and W. Sun, "A novel bandwidth allocation scheme for elastic and inelastic services in peer-to-peer networks," *IAENG International Journal of Computer Science*, vol. 46, no. 2, pp. 163–169, 2019.

[7] T. S. Alemayehu, J. H. Kim, and W. D. Cho, "Optimal replacement model for the physical component of safety critical smart-world CPSs," *Journal of Ambient Intelligence and Humanized Computing*, no. 13, pp. 4579–4590, 2022.

[8] H. J. Xuan, X. L. Zhao, J. W. Fan, Y. H. Xue, F. F. Zhu, and Y. L. Li, "VNF service chain deployment algorithm in 5G communication based on reinforcement learning," *IAENG Internaitonal Journal of Computer Science*, vol. 48, no. 1, pp. 1–7, 2021.

[9] H. Hantouti, N. Benamar, and T. Taleb, "Service function chaining in 5G and beyond networks: Challenges and open research issues," *IEEE Network*, vol. 34, no. 4, pp. 320–327, 2020.

[10] V. Eramo, F. G. Lavacca, T. Catena, and P. J. P. Salazar, "Application of a long short term memory neural predictor with asymmetric loss function for the resource allocation in NFV network architectures," *Computer Networks*, vol. 193, pp. 108 104–108 116, 2021.

[11] H. T. Zhang, G. F. Wu, and W. S. Bu, "Research of agent based control model for embedded networked system," *Engineering Letters*, vol. 27, no. 2, pp. 278–286, 2019.

[12] H. J. Xuan, J. Lu, N. Li, and L. J. Wang, "Novel virtual network function service chain deployment algorithm based on Q-learning," *IAENG International Journal of Computer Science*, vol. 50, no. 2, pp. 736–744, 2023.

[13] H. A. Malhat, A. S. Zainud-Deen, M. Rihan, and M. M. Badway, "Elements failure detection and radiation pattern correction for time-modulated linear antenna arrays using particle swarm optimization," *Wireless Personal Communications*, vol. 125, no. 3, pp. 2055–2073, 2022.

[14] L. Liu, H. Hu, Y. Luo, and Y. Wen, "When wireless video streaming meets AI: A deep learning approach," *IEEE Wireless Communications*, vol. 27, no. 2, pp. 127–133, 2020.

[15] N. Zhang, P. Yang, S. Zhang, D. Chen, W. Zhuang, B. Liang, and X. S. Shen, "Software defined networking enabled wireless network virtualization: Challenges and solutions," *IEEE Network*, vol. 31, no. 5, pp. 42–49, 2020.

[16] C. Pham, N. H. Tran, S. Ren, W. Saad, and C. S. Hong, "Traffic-aware and energy-efficient VNF placement for service chaining: Joint sampling and matching approach," *IEEE Transactions on Services Computing*, vol. 13, no. 1, pp. 172–185, 2020.

[17] P. Kumar, S. Shilpi, A. Kanungo, V. Gupta, and N. K. Gupta, "A novel ultra wideband antenna design and parameter tuning using hybrid optimization strategy," *Wireless Personal Communications*, no. 122, pp. 1129–1152, 2013.

[18] S. Araujo, F. Souza, and G. R. Mateus, "A hybrid optimization-machine learning approach for the VNF placement and chaining problem," *Computer Networks*, vol. 199, p. 108474, 2021.

[19] A. Khatiri, G. Mirjalily, and Z. Q. Luo, "Balanced resource allocation for vnf service chain provisioning in inter-datacenter elastic optical networks," *Computer Networks*, vol. 203, no. 11, pp. 1–12, 2022.

[20] Z. Xu, Z. Zhang, W. Liang, Q. Xia, and G. Wu, "Qos-aware VNF placement and service chaining for IoT applications in multi-tier mobile edge networks," *ACM Transactions on Sensor Networks*, vol. 16, no. 3, pp. 1–27, 2020.

[21] L. Y. Xu, A. Y. Deng, M. Z. Ge, J. L. Shi, and G. Y. Gao, "A MADM-based handover management in software-defined 5G network," *Engineering Letters*, vol. 27, no. 4, pp. 842–849, 2019.

[22] E. Guler, S. Devaraju, G. Luo, L. Tian, and X. Cao, "Multicast-aware service function tree embedding," in *2019 IEEE 20th International Conference on High Performance Switching and Routing (HPSR)*, 2019, pp. 1–6.

[23] B. Ren, D. Guo, Y. Shen, G. Tang, and X. Lin, "Embedding service function tree with minimum cost for NFV-enabled multicast," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 5, pp. 1085–1097, 2019.

[24] Z. Xu, W. Liang, M. Huang, M. Jia, S. Guo, and A. Galis, "Efficient nfv-enabled multicasting in SDNs," *IEEE Transactions on Communications*, vol. 67, no. 3, pp. 2052–2070, 2018.

[25] O. Alhussein, P. T. Do, Q. Ye, J. Li, and J. Rao, "A virtual network customization framework for multicast services in NFV-enabled core networks," pp. 1025–1039, 2020.

[26] N. Kiji, T. Sato, R. Shinkuma, and E. Oki, "Virtual network function placement and routing for multicast service chaining using merged paths," *Optical Switching and Networking*, vol. 36, p. 100554, 2020.

[27] M. Asgarian, G. Mirjalily, and Z. Q. Luo, "Embedding multicast service function chains in NFV-enabled networks," *IEEE Communications Letters*, vol. 4, no. 25, pp. 1264–1268, 2021.

[28] S. Thabo, K. Xie, X. Zhou, S. Hemant, and Z. Li, "Delay bounded multi-source multicast in software-defined networking," *Electronics*, vol. 7, no. 1, p. 10, 2018.

[29] K. Xie, X. Zhou, T. Semong, and S. He, "Multi-source multicast routing with QoS constraints in network function virtualization," in *2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–8.

[30] A. Muhammad, I. Sorkhoh, L. Qu, and C. Ass, "Delay-sensitive multi-source multicast resource optimization in NFV-enabled networks: A column generation approach," vol. 18, no. 1, 2021, pp. 286–300.

[31] Y. Wu and J. Zhou, "Dynamic service function chaining orchestration in a multi-domain: A heuristic approach based on SRv6." *Sensors*, vol. 21, no. 19, p. 6563, 2021.

[32] S. Guo, Y. Qi, Y. Jin, W. Li, and L. Meng, "Endogenous trusted drl-based service function chain orchestration for IoT," *IEEE Transactions on Computers*, vol. 71, no. 2, pp. 397–406, 2021.

[33] S. R. Zahedi, S. Jamali, and P. Bayat, "A power-efficient and performance-aware online virtual network function placement in SDN/NFV-enabled networks," *Computer Networks*, vol. 205, p. 108753, 2022.

[34] K. Zahoor, K. Bilal, A. Erbad, A. Mohamed, and M. Guizani, "Multicast at edge: An edge network architecture for service-less crowdsourced live video multicast," *IEEE Access*, vol. 9, pp. 59 508–59 526, 2021.

[35] Q. Long and C. Assi, "Reliability-aware multi-source multicast hybrid routing in softwarized networks," *IEEE Access*, vol. 8, pp. 113 331–113 341, 2020.

[36] H. Xing, X. Zhou, X. Wang, S. Luo, P. Dai, K. Li, and H. Yang, "An integer encoding grey wolf optimizer for virtual network function placement," *Applied Soft Computing*, vol. 76, pp. 575–594, 2019.