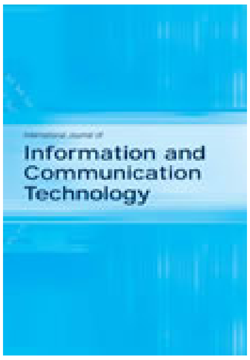
INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research



International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Resource backup algorithm of service function chain based on network characteristics and sharing advantages

Shunli Zhang

Article History:

Received:	24 May 2023
Last revised:	02 September 2023
Accepted:	08 December 2023
Published online:	03 May 2024

Resource backup algorithm of service function chain based on network characteristics and sharing advantages

Shunli Zhang

Jinzhong University,
No. 199 Wenhua Street, Yuci District, Jinzhong City,
Shanxi Province, China
Email: 544891117@qq.com

Abstract: In the context of network function virtualisation (NFV), the issue of low reliability during deployment of service function chain (SFC) has become an urgent and important issue to be addressed. A heuristic advantage node selection and backup algorithm were proposed when selecting backup underlying nodes for VNF instances. The advantages of the underlying nodes as alternative nodes for VNF instances based on their resource availability, traffic size, and recovery ability are evaluated. K-kernel decomposition algorithm was used to identify core nodes and edge nodes. The simulation results show that the algorithm proposed in this paper reduces the consumption of backup instances and bandwidth resources while improving the reliability of SFC and improves the availability and acceptance rate of SFC. This algorithm determines the optimisation order of VNF instances based on the superiority evaluation results of VNF instances and optional backup underlying nodes, improving the utilisation of backup resources.

Keywords: network function virtualisation; NFV; service function chain; SFC; service reliability; resource backup; feature aware.

Reference to this paper should be made as follows: Zhang, S. (2024) 'Resource backup algorithm of service function chain based on network characteristics and sharing advantages', *Int. J. Information and Communication Technology*, Vol. 24, No. 5, pp.36–53.

Biographical notes: Shunli Zhang received his MS in Computer Science from the Taiyuan University of Technology, China, and PhD in Computer Science from the Beijing University of Posts and Telecommunications in 2012. He is currently an Associate Professor at Jinzhong University, Shanxi, China. He has authored journal and conference papers about the resource management and fault diagnosis of network. His research interests include management for virtual networks and next generation networks.

1 Introduction

With the rapid development and application of 5G technology, network function virtualisation (NFV) has been proposed and rapidly developed (Magoula et al., 2021). In the NFV environment, network functions can be decoupled from the hardware environment, significantly improving the efficiency and flexibility of network function deployment and reducing the construction and deployment costs of the network (Zhu et al., 2022). Network functions are deployed on the underlying network nodes using virtual network functions (VNFs). Service requests are deployed using the service function chain (SFC) approach (Liu et al., 2022; Guo et al., 2021). The literature Cotroneo et al. (2014) investigates issues related to NFV reliability and shows that software failures in the VNF itself and hardware failures in the underlying network nodes that host it can lead to VNF unavailability. Software failures can come from the VNF itself or from misconfiguration of virtual machines. Hardware failures can come from the failure of the server hosting the VNF. When a VNF is unavailable, the SFC of which it is a part also suffers from performance degradation or unavailability. Therefore, how to improve the availability of VNFs is an important research element for service continuity. Research results on improving the availability of VNFs can be divided into three types: estimating the availability of nodes in advance, recovery mechanisms for failed VNFs, and resource backup mechanisms.

In terms of research on estimating the availability of nodes in advance, the literature He et al. (2019), Zhu et al. (2020), Bangqiang et al. (2016) estimates the availability of nodes in advance by analysing the usage records or maintenance records of network equipment, and designs various optimised placement models to offset the effects of unavailability. Literature Sun et al. (2018) obtains the reliability of the underlying nodes by analysing network management centre data and designs resource deployment algorithms for maximising SFC reliability. The literature He et al. (2019), Zhu et al. (2020), Bangqiang et al. (2016), Sun et al. (2018) is identical in that the availability of the nodes is estimated in advance based on network operation data, and then the VNF is deployed using a resource optimisation strategy that enhances SFC reliability. However, SFC outages can still occur when the underlying node fails. To address the problem of SFC outages when the underlying node fails, studies have proposed recovery mechanisms for failed VNFs. The literature Soualah et al. (2017) uses a virtual link remapping strategy to deploy an available new underlying path for the SFC affected by the failure. The literature Latchoumy and Khader (2015) predicts process failures during user job scheduling based on historical data and performs rapid recovery. The recovery mechanism of a failed VNF allows for fast fault recovery in case of VNF failure; however, fault recovery inevitably leads to SFC outages and degrades the quality of service.

The resource backup mechanism is one of the efficient measures to improve the availability of SFC as opposed to estimating the availability of nodes in advance and the recovery mechanism of failed VNFs. The resource backup problem of SFC adopts a strategy of reserving network resources in advance for critical or special resources, which can significantly improve the reliability of the network. The literature Sato et al. (2018) proposes a VM state monitoring strategy to provide backup resources for VMs when their state is unavailable. The literature Anthoniraj and Saraswathi (2018) uses backup resources to back up the fault resources predicted by intelligent monitoring algorithms. The literature Sato et al. (2018), Anthoniraj and Saraswathi (2018) backs up unreliable resources, which improves the availability of SFCs but also increases the overhead of backing up resources rapidly. To reduce the backup resource overhead, the literature Kang et al. (2021) defers the start-up time of the backup resources and reduces the backup resource overhead. Literature Yang et al. (2021), Zhai et al. (2020) uses network features to aggregate VNFs and physical nodes, which improves the utilisation of network resources. Literature Cai et al. (2021) uses multi-path routing and hierarchical graphs to solve the problem of low SFC reliability in disaster zones. Literature Qu et al. (2017) used integer linear programming to solve the optimal backup strategy and reduce the overhead of backup resources. The literature Yang et al. (2021), Zhai et al. (2020), Cai et al. (2021), Qu et al. (2017) used various optimisation measures to reduce the overhead of backup resources. However, these studies adopt a backup strategy that analyses and solves for minimising the overhead on an SFC-by-SFC basis and do not explore the resource sharing relationship between SFCs. To solve this problem, the literature Zhang et al. (2019), Wang et al. (2021) aggregates multiple SFCs into a service function diagram (SFG) and performs reliability filtering on the SFG set. When backing up the SFCs whose reliability cannot meet the requirements, priority is given to backing up resources for the VNFs with greater centrality. Although the use of SFGs improves the efficiency of backups. However, the resource backup does not consider the underlying network resource characteristics, nor does it analyse and perform resource backup based on the sharing relationship between SFCs and VNF instances that need to be backed up, and the sharing of backup resources needs to be further improved.

In order to reduce the cost of backup resources, existing research has focused on the reliability of individual SFCs using artificial intelligence algorithms, lacking in-depth analysis of the characteristics of underlying network resources and the relationship between shared resources between VNF instances. The author has been researching in the field of network virtualisation resource management for many years, and commonly used research methods include intelligent optimisation algorithms and network feature-based optimisation algorithms. However, most of the better network resource management algorithms excavate the relevant features of network resources, thereby significantly improving the performance of resource management algorithms. In order to improve the utilisation of underlying network resources while meeting the reliability requirements of SFC resources, this paper proposes a SFC resource backup algorithm by mining network features and resource sharing advantages. A heuristic advantage node selection and backup algorithm were proposed when selecting backup underlying nodes for VNF instances. By analysing the results of comparing the algorithm in this paper with the classical algorithms in current research, it can be seen that the algorithm in this paper saves the resource consumption of backup instances and backup bandwidth, and further improves the availability and acceptance rate of SFC.

2 Problem description

2.1 Network models

In an NFV environment, the undirected graph $G_s = (N_s, L_s)$ is used to represent the underlying network. Use $n_s^i \in N_s$ to represent the underlying nodes. Use $l_s^j \in L_s$ to represent the underlying link. The attributes of the underlying node $n_s^i \in N_s$ include compute capacity $c_{n_s^i}$, the type $t_{n_s^i}$ of VNFs that can be deployed, hardware reliability $r_{n_s^i}$, and location $l_{n_s^i}$. The underlying link $l_s^j \in L_s$ attribute is the amount of link bandwidth resources $b_{l_s^j}$. Use $G_v(N_v, E_v)$ to denote a set SFG consisting of K SFCs, where N_v denotes the set of VNF instance nodes in the SFG, and E_v denotes the virtual link between VNF instance nodes.

The SFC is represented using a directed acyclic graph $G_{sfc} = (VNF_{sfc}, DL_{sfc}, R_{sfc})$. VNF_{sfc} represents the set of VNF instances, DL_{sfc} represents the dependencies between VNF instances in the SFC, and R_{sfc} represents the reliability requirements of the SFC, expressed using the minimum value of the probability of normal operation of the SFC, taking values in the range [0, 1]. The resources of a VNF instance $vnf_k \in VNF_{sfc}$ are allocated by the underlying node, and its attributes include the type t_{vnf_k} of VNF, the computational resource requirement c_{vnf_k}, the outflow bandwidth demand b_{vnf_k}, and reliability r_{vnf_k}.

The value r_{vnf_k} of reliability is influenced by the reliability of the underlying node and software. As software reliability is less related to network characteristics, this paper focuses on the impact of underlying node reliability on the reliability of VNF instances. Use $DL_{vnf_k} \in DL_{sfc}$ to denote the dependency constraint posed by the connection order between VNF instances in an SFC. For example, in an SFC sequential connectivity relationship, VNF instance vnf_2 comes after vnf_1. $DL_{vnf_1} > DL_{vnf_2}$ can be used to denote that instance vnf_2 depends on instance vnf_1. After deploying VNF instances to the underlying nodes, it is necessary to determine whether the reliability of the SFC meets the required R_{sfc}. The true reliability R_{sfc}^{real} of the SFC is expressed using the reliability r_{vnf_k} of all the VNF instances it contains, calculated using equation (1), where n denotes the number of VNF instances contained in the SFC.

$$R_{sfc}^{real} = \prod_{k=1}^{n} r_{vnf_k} \tag{1}$$

when $R_{sfc}^{real} < R_{sfc}$, it indicates that the real reliability R_{sfc}^{real} of SFC is lower than the required reliability R_{sfc}. At this time, the VNF instance reliability needs to be optimised to ensure that the real reliability of SFC meets the requirements, thus ensuring the normal operation of the business on SFC. When the underlying node fails or an external event causes the underlying node to be unavailable, the reliability of the VNFs carried on the underlying node is reduced, resulting in the reliability of the SFC not meeting the customer's requirements. Therefore, before the SFC provides services to customers, it is necessary to quickly improve the reliability of the SFC by means of a resource backup strategy until it meets the customer's requirements.

2.2 Resource backup policy

Assume that the reliability of the VNF instance vnf_k to be backed up is denoted using r_{vnf_k} and the reliability of the backup underlying node used is denoted using r'_{vnf_k}. At this point, the reliability $r_{vnf_k}^{b}$ of the VNF instance vnf_k after backup can be calculated using equation (2).

$$r_{vnf_k}^{b} = 1 - \left(1 - r_{vnf_k}\right)\left(1 - r'_{vnf_k}\right) \tag{2}$$

The objective function for the SFC reliability optimisation problem is shown in equation (3). Where $L_{vnf_i}^{pre}$ and $L_{vnf_i}^{post}$ denote the length of the new underlay path that needs to be created between the backup underlay node and the underlay node of the previous and next VNF instance respectively, and b_{pre} and b_{vnf} denote the bandwidth flowing out of the previous VNF of the VNF instance and VNF instance respectively. The constraints on the objective function shown in equation (3) are shown in equations (4–6). In equation (4), $x_{vnf_i}^{n_s^j} \in \{0,1\}$ indicates whether the VNF instance is mapped to the underlying node n_s^j, and $b_{vnf_i}^{n_s^j} \in \{0,1\}$ indicates whether the backup node of the VNF instance is mapped to the underlying node n_s^j.

Therefore, the constraint in equation (4) indicates that the mapped underlying node of the VNF instance and the backup underlying node to be mapped cannot be the same underlying node. This constraint prevents both underlying nodes of the VNF instance from being unavailable at the same time. Equation (5) indicates that the reliability of the SFC after the backup of the VNF instance should meet the reliability requirements of the SFC. Equation (6) indicates that the number of resources on the underlying node n_s^j used by the backup instance cannot be greater than the number $c_{n_s^j}$ of resources available on the underlying node n_s^j. α_{vnf_i} denotes the number of resources demanded by vnf_i.

$$\min \sum_{vnf_i} \left(L_{vnf_i}^{pre} \cdot b_{pre} + L_{vnf_i}^{post} \cdot b_{vnf_i} \right) \tag{3}$$

$$x_{vnf_i}^{n_s^j} + b_{vnf_i}^{n_s^j} \le 1 \tag{4}$$

$$R_{sfc}^{real} \ge R_{sfc} \tag{5}$$

$$\sum_{vnf_i} \alpha_{vnf_i} \cdot b_{vnf_i}^{n_s^j} \le c_{n_s^j} \tag{6}$$

3 Network characterisation

3.1 Network characteristics of the underlying nodes

3.1.1 Resource availability of the underlying nodes

The resource utilisation $avail(n_s^i)$ of the bottom node n_s^i is calculated using equation (7). $CPU_{all}(n_s^i)$ denotes the total amount of computing capacity of the bottom node n_s^i. $CPU_{avail}(n_s^i)$ denotes the available computing capacity of the underlying node n_s^i. $BW_{all}(n_s^i)$ denotes the total amount of bandwidth resources of the bottom node n_s^i. $BW_{avail}(n_s^i)$ denotes the amount of available bandwidth resources of the bottom node n_s^i. $e_s^j \in E(n_s^i)$ denotes the bottom link in the set of links $E(n_s^i)$ connected to the bottom node n_s^i. $BW(e_s^i)$ denotes the amount of bandwidth resources of the bottom link e_s^j. $BW_{avail}(e_s^i)$ denotes the amount of available bandwidth resources of the underlying link e_s^j. When the resource utilisation of the bottom node is high, the bottom node is prone to failure. Also, when the bottom node has more free resources, it may be selected as a backup resource.

$$avail(n_s^i) = \frac{CPU_{avail}(n_s^i)}{CPU_{all}(n_s^i)} + \frac{BW_{avail}(n_s^i)}{BW_{all}(n_s^i)} \tag{7}$$

$$BW_{all}(n_s^i) = \sum\nolimits_{e_s^j \in E(n_s^i)} BW(e_s^j) \tag{8}$$

$$BW_{avail}(n_s^i) = \sum\nolimits_{e_s^j \in E(n_s^i)} BW_{avail}(e_s^j) \tag{9}$$

3.1.2 Traffic size of the underlying node

The traffic size of the bottom node n_s^i determines the size of the business carried on the bottom node. When the number of services carried on the bottom node is large, the bottom node is an important node. The set of directly connected nodes of the bottom node n_s^i is denoted by $EC(n_s^i)$. The virtual link carried on the link e_s^{ij} between the underlying node n_s^i and the underlying node $n_s^j \in EC(n_s^i)$ is denoted using e_{ij}^V. $f_{ij}^{e^V}$ denotes the traffic on link e_s^{ij} between the bottom node n_s^i and the bottom node $n_s^j \in EC(n_s^i)$. $f_{ij}^{e^V} > 0$ indicates that a virtual link is carried on the underlying link and is represented using a decision variable δ_{ij} greater than 0. Otherwise, the decision variable δ_{ij} is set to 0. The traffic $p_{n_s^{ij}}$ from the bottom node n_s^j to the bottom node n_s^i is calculated using equation (10).

$$p_{n_s^j} = \delta_{ij} f_{ij}^{e^V} \tag{10}$$

All traffic $p_{n_s^i}^{sum}$ of the directly connected nodes of the bottom node n_s^i is calculated using equation (11). Where, the larger the value of $p_{n_s^i}^{sum}$ takes, the larger the service traffic of SFC carried on the current node, and the richer the link resources of the node to other nodes.

$$p_{n_s^i}^{sum} = \sum_{n_s^j \in EC(n_s^i)} p_{n_s^{ij}} \tag{11}$$

3.1.3 Resilience of the underlying nodes

The resilience of the bottom node n_s^i refers to how long the bottom node can recover after a failure. When the number of surrounding resources of the bottom node is high, the current bottom node has more backup resources. At this time, the current bottom node has a higher recovery capability. The recovery capability of the bottom node n_s^i is expressed using $resil(n_s^i)$ and calculated using equation (12). Where $\varphi(n_s^i)$ denotes the set consisting of the bottom nodes that have direct links to the bottom node n_s^i. α and β denote the weight factors of computational resources and link resources of node, respectively.

$$resil(n_s^i) = \sum\nolimits_{n_s^j \in \varphi(n_s^i)} \alpha \cdot CPU_{avail}(n_s^j) + \beta \cdot BW_{avail}(n_s^j) \tag{12}$$

The resource availability of the bottom node describes the strength of the reliability of the bottom node. The traffic size of the bottom node describes the size of the service carried. The resilience of the bottom node describes the ease of backing up resources of the bottom node and the ability to save resources for the bottom link when backing up. The three attributes of the bottom node are normalised to the maximum and minimum using equation (13). Where x denotes the value of an attribute of the bottom node, x_{mas} denotes the maximum value of the attribute data of that class, x_{min} denotes the minimum value of the attribute data of that class, and x' denotes the normalised value of an attribute of the bottom node.

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{13}$$

The superiority of each bottom node as an alternative bottom node, referred to as the backup superiority of the bottom node, can be obtained by linearly transforming the three attribute values of each bottom node n_s^i to the range [0,1] and then summing them, using the $BA(n_s^i)$ representation.

3.2 Network characteristics of VNF instances

The network characteristics of VNF instances are mainly used to determine their advantages as an alternative. In this paper, SFG network model is used to select VNF backup instances. In the SFG network model, the selected VNF backup instance has more VNF instances associated in the SFG if it has higher degree and bandwidth resources, and its selected backup underlying node can be shared with other VNF instances. The superiority of the VNF instance as an alternative object is expressed using $ADV(vnf_k)$, which is calculated using equation (14). Where λ and η denote the balance adjustment factor of node degree and node bandwidth traffic, respectively, and $dg_{vnf_k}^{in}$ and $dg_{vnf_k}^{out}$ denote the number of incoming degrees and outgoing degrees of VNF instance vnf_k. $bw_{vnf_k}^{in}$ and $bw_{vnf_k}^{out}$ denote the inflow bandwidth and outflow bandwidth values of VNF instance vnf_k, respectively.

$$ADV\left(vnf_k\right) = \lambda dg_{vnf_k}^{in} \cdot \eta bw_{vnf_k}^{in} + \lambda dg_{vnf_k}^{out} \cdot \eta bw_{vnf_k}^{out} \tag{14}$$

Based on the advantages of VNF instances as alternative objects, this article proposes the concept of resource sharing advantages for underlying nodes. The advantage of resource sharing among underlying nodes refers to their ability to be shared by multiple VNF instances as backup nodes. In evaluating the resource sharing advantages of underlying nodes, the number of VNF instances that can be restored by the current underlying node and the superiority value of VNF instances are used to measure. When the number of VNF instances that the underlying node can recover is large and the superiority value of VNF instances is high, it indicates that the current underlying node has a significant sharing advantage. In the SFC resource backup algorithm, selecting the underlying node with a large number of recoverable VNF instances and the highest superiority of VNF instances as the backup underlying node can significantly improve the utilisation rate of backup underlying node resources.

4 Algorithm

4.1 Reliability optimisation algorithm

To solve the problem of low reliability during SFC deployment, the resource backup algorithm of SFC based on network characteristics and sharing advantages (SFCRBAoNCSA) proposed in this paper is shown in Table 1.

The algorithm consists of seven steps: constructing the set of available alternative underlying nodes and calculating the node superiority feature values, constructing the set of underlying nodes of VNF instances that need to be backed up, constructing the set of nodes to be backed up, deleting the nodes that cannot meet the resource requirements, performing resource backup, updating the reliability of the relevant SFCs, and judging whether the reliability of all SFCs meets the requirements.

Step 1 Is to construct a set of available alternative underlying nodes and calculate the node superiority feature values. Considering that the backup node has good network performance (such as sufficient resources and high reliability), and the underlying nodes require more computational overhead to calculate their network characteristics, the K-kernel algorithm is first used to construct the core node set N_s^{BD}. The k-kernel decomposition algorithm is an analytical tool for quickly identifying special subnetworks (Dorogovtsev et al., 2006; Alvarez-Hamelin et al., 2005; Qing et al., 2012). Using an iterative algorithm to delete nodes and edges with degree 1 in the topology, set the deleted nodes as 1-core nodes. Using an iterative algorithm to delete nodes and edges with degree 2, set the deleted nodes as 2-core nodes. Repeat until all nodes are assigned a kernel index. Finally, mark 1 core node as the edge network element, and mark nodes larger than 1 core as the core network element. Secondly, use equations (7), (11), and (12) to calculate the $avail(n_s^i)$, $p_{n_s^i}^{sum}$, and $resil(n_s^i)$ of the underlying node $n_s^i \in N_s^{BD}$. Normalise the three attributes of the underlying node using equation (13), and sum them to obtain the backup superiority $BA(n_s^i)$ of the underlying node.

Step 2 Is to build the underlying node collection of VNF instances that need to be backed up. Firstly, search for $sfc_{not-rel}^i$ whose true reliability cannot meet the reliability requirements, forming the set $SFC_{not\text{-}rel}$. Secondly, form a set $VNF_{SFC_{not-rel}}$ of all VNF instances contained in $SFC_{not\text{-}rel}$, and use equation (14) to calculate the superiority $ADV(vnf_k)$ of each VNF instance vnf_k in $VNF_{SFC_{not-rel}}$. Finally, based on the resource allocation relationship, find the underlying node $n_s^{vnf\,j}$ of instance $vnf_{SFC_{not-rel}^i}^j \in VNF_{SFC_{not-rel}^i}$ in $SFC_{not\text{-}rel}$ to form a set N_s^{VNF-b}.

Step 3 Is to build a set of nodes to be backed up. Firstly, find the nodes $n_s^{vnf\,j} \in N_s^{VNF-b}$ contained in the geographic location range δ of each node in $n_s^k \in N_s^{BD}$, and delete the nodes that do not satisfy the constraints of equation

(4) to obtain the set $N_{n_s^k}^{n_{VNF-b}^{\delta}}$. The number of underlying nodes contained in $N_{n_s^k}^{n_{VNF-b}^{\delta}}$ is $number_{n_s^k}^{VNF-b}$. Secondly, put the nodes of $number_{n_s^k}^{VNF-b} \geq 1$ in $N_{n_s^k}^{n_{VNF-b}^{\delta}}$ into the set N_s^{toBU} of nodes to be backed up. Judge whether N_s^{toBU} is empty, if it is empty, increase the geographic location range δ by 1 unit and return to (1) in this step.

Step 4 Is to delete the nodes that cannot meet the resource requirements. Firstly, put the node of $number_{n_s^k}^{VNF-b} > 3$ in the collection $N_s^{toBU-more3}$, and select the 3 nodes in $N_s^{toBU-more3}$, that are closest to $n_s^k \in N_s^{toBU}$. Secondly, delete node that does not meet resource needs in N_s^{toBU}, and get a new collection N_s^{toBU-U}.

Step 5 Is to perform a resource backup. This step requires the invocation of heuristic advantage node selection and backup algorithms, as detailed in the next section.

Step 6 Is to update the reliability of the relevant SFCs. Firstly, find the underlying node set $N_{n_s^k}^{n_{VNF-b}^{\delta}}$ of the backup node for the VNF in the set N_s^{VNF-b}. Secondly, find the SFC containing the VNF instance corresponding to set $N_{n_s^k}^{n_{VNF-b}^{\delta}}$ in $SFC_{not\text{-}rel}$, and calculate the SFC reliability in parallel. If SFC meets the requirements, delete the current SFC from $SFC_{not\text{-}rel}$, and delete the underlying node of the VNF instance of the SFC in N_s^{VNF-b}.

Table 1 Algorithm SFCRBAONCSA

1	Constructing the set of available alternative underlying nodes and calculating the node superiority feature values	
	a	Construct the core set N_s^{BD} of nodes using the K-core method
	b	Calculate $avail(n_s^i)$, $p_{n_s^i}^{sum}$, $resil(n_s^i)$ of the underlying node $n_s^i \in N_s^{BD}$ using equations (7), (11), and (12), respectively
	c	Use equation (13) to perform normalisation on the three attributes of the bottom node, and obtain the backup superiority $BA(n_s^i)$ after summing
2	Constructing the set of underlying nodes of VNF instances that need to be backed up	
	a	Find $sfc_{not-rel}^i$ whose true reliability does not meet reliability requirements, which constitutes the set $SFC_{not-rel}$
	b	All VNF instances in $SFC_{not-rel}$ form a set $VNF_{SFC_{not-rel}}$
	c	Use equation (14) to calculate the superiority $ADV(vnf_k)$ of each VNF instance vnf_k in $VNF_{SFC_{not-rel}}$
	d	Find the underlying node $n_s^{vnf\,j}$ of the instances $vnf_{SFC_{not-rel}^i}^j \in VNF_{SFC_{not-rel}^i}$ in according to the resource allocation relationship, which constitutes the set N_s^{VNF-b}

Table 1 Algorithm SFCRBAONCSA (continued)

3	Constructing the set of nodes to be backed up	
	a	Find the nodes $n_s^{vnf\,j} \in N_s^{VNF-b}$ contained in the geographic location range δ of each node in $n_s^k \in N_s^{BD}$, and delete the nodes that do not satisfy the constraints of equation (4) to obtain the set $N_{n_s^k}^{n_{VNF-b}^{\delta}}$. The number of underlying nodes contained in $N_{n_s^k}^{n_{VNF-b}^{\delta}}$. is $number_{n_s^k}^{VNF-b}$
	b	Put the nodes of $number_{n_s^k}^{VNF-b} \geq 1$ in $N_{n_s^k}^{n_{VNF-b}^{\delta}}$ into the set N_s^{toBU} of nodes to be backed up
	c	Judge whether N_s^{toBU} is empty, if it is empty, increase the geographic location range δ by 1 unit and return to (1) in this step
4	Deleting the nodes that cannot meet the resource requirements	
	a	Put the node of $number_{n_s^k}^{VNF-b} > 3$ in the collection $N_s^{toBU-more3}$
	b	Select the 3 nodes in $N_s^{toBU-more3}$ that are closest to $n_s^k \in N_s^{toBU}$
	c	Delete node that does not meet resource needs in N_s^{toBU}, and get a new collection N_s^{toBU-U}
	d	If N_s^{toBU-U} is empty, return step 3
5	Performing resource backup (see Table 2)	
6	Updating the reliability of the relevant SFCs	
	a	Find the underlying node set $N_{n_s^k}^{n_{VNF-b}^{\delta}}$ of the backup node for the VNF in the set N_s^{VNF-b}
	b	Find the SFC containing the VNF instance corresponding to set $N_{n_s^k}^{n_{VNF-b}^{\delta}}$ in $SFC_{not\text{-}rel}$, and calculate the SFC reliability in parallel
	c	If SFC meets the requirements, delete the current SFC from $SFC_{not\text{-}rel}$, and delete the underlying node of the VNF instance of the SFC in N_s^{VNF-b}
7	Judging whether the reliability of all SFCs meets the requirements	
	Judge whether $SFC_{not\text{-}rel}$ is empty. If it is empty, the algorithm ends. Otherwise, return to step 3.	

4.2 Heuristic superior node selection and backup algorithm

In order to select the best resource from the bottom nodes for backup, this paper proposes a heuristic algorithm for selecting the superior node and backup (see Table 2). The algorithm selects the bottom node with the largest number of recoverable VNF instances and the greatest advantages as the backup bottom node.

The algorithm includes three parts: calculating the resilience of the underlying nodes and arranging them in descending order (Steps 1–2), using the underlying node with the greatest advantage as the backup node (Steps 3–4), and allocating link resources to the front and rear nodes of the underlying node of the VNF instance that gets the backup resources (Steps 5–6).

- In step 1, the recovery capability of bottom layer node $n_s^k \in N_s^{toBU-U}$ is calculated using equation (15) to represent the sum of the advantages of all the virtual network instances it recovers. Where, $number_{n_s^k}^{VNF-b}$ represents the number of VNF instances that can be recovered by the underlying node $n_s^k \in N_s^{toBU-U}$.
- In step 2, arrange the nodes in set N_s^{toBU-U} in descending order of resilience. In step 3, select the first q bottom nodes in N_s^{toBU-U}. In step 4, select the bottom node with the greatest advantage as the backup bottom node, and allocate backup resources according to the maximum demand in set $N_{n_s^k}^{n_{VNF-b}^{\delta}}$. In step 5, to achieve network connectivity, find the previous and next underlying nodes of the underlying node of each VNF instance in $N_{n_s^k}^{n_{VNF-b}^{\delta}}$, and put them into the collection $N_{n_s^k}^{link}$ to be connected. In step 6, connect the current bottom node n_s^k with each bottom node in $N_{n_s^k}^{link}$ using the shortest path algorithm.

$$\text{Resil}(n_s^k) = \sum_{t=1}^{number_{n_s^k}^{VNF-b}} ADV\left(vnf_k^t\right) \tag{15}$$

Table 2 Heuristic superior node selection and backup algorithm

1	Calculate the resilience of each node $n_s^k \in N_s^{toBU-U}$ in set N_s^{toBU-U} using equation (15)
2	Arrange the nodes in set N_s^{toBU-U} in descending order of resilience
3	Select the first q bottom nodes in N_s^{toBU-U}
4	Select the bottom node with the greatest advantage as the backup bottom node, and allocate backup resources according to the maximum demand in set $N_{n_s^k}^{n_{VNF-b}^{\delta}}$
5	Find the previous and next underlying nodes of the underlying node of each VNF instance in $N_{n_s^k}^{n_{VNF-b}^{\delta}}$, and put them into the collection $N_{n_s^k}^{link}$ to be connected
6	Connect the current bottom node n_s^k with each bottom node in $N_{n_s^k}^{link}$ using the shortest path algorithm

5 Performance analysis

5.1 Experimental environment

The network topology in the experimental environment is generated using GT-ITM topology generator (Zegura et al., 1996; GT-ITM, 2000). The underlying network topology is randomly extracted from a square area with 100 edges. The number of bottom nodes is 100, and the bottom link is obtained by connecting the bottom nodes randomly with a probability of 0.5. In terms of network topology generation for SFC requests, the number of NFV instance nodes is evenly distributed from 2 to 6. The CPU resources of the underlying node and the bandwidth of the underlying link follow a uniform distribution of 50 to 100 units. The CPU resources and link bandwidth on the NFV instance nodes follow a uniform distribution of 1 to 15. The reliability of the underlying node is described by a random value from 0.9 to 0.999. Each underlying node can support three types of VNF instances. When underlying node hosts one type of VNF instance, it cannot host other types of VNF instances.

In order to evaluate the performance and resource efficiency of the proposed backup methods, the classical algorithms in the current research are selected for comparison. The classic algorithm selected is node ranking algorithm with centrality and reliability (NRCR) (Wang et al., 2021). This algorithm backs up VNF instance resources according to centrality and reliability, but it does not consider the backup advantages of underlying node and the network characteristics of resource sharing of VNF instance. To analyse the algorithm performance when network characteristics of the VNF instance backup resource sharing are not considered, the reliability optimisation algorithm of SFC based on distance and superiority (ROAoDS) is selected for comparison (Yang et al., 2021; Zhai et al., 2020). After finding the underlying node corresponding to the NFV instance that need to be backed up urgently, the algorithm ROAoDS selects the underlying node that is closer to the underlying node, meets resource requirements and has greater backup advantages as the backup node. Compared to the first two comparative algorithms, this paper proposes algorithm SFCRBAONCSA, which considers both the backup superiority attribute of underlying node and the network characteristics of the backup resource sharing of VNF instance. So, the algorithm in this article is an optimisation and supplement to the first two comparative algorithms.

In terms of quantitative indicators for performance analysis, analyse from three dimensions: backup resource consumption, SFC availability, and SFC acceptance rate. In terms of backup resource consumption, considering that the number of backup instances and backup bandwidth resource consumption can reflect the consumption of underlying network resources, two dimensions of backup instance number and backup bandwidth resource consumption are used to analyse backup resource consumption.

In terms of SFC availability, the main purpose is to evaluate whether reliability optimisation algorithms can improve the reliability of SFC. The availability rate of SFC refers to the proportion of the number of SFCs that are normally available to the total number of SFCs after a failure of the underlying node. This indicator can analyse the level of improvement in SFC reliability by backup strategies. When all the underlying nodes corresponding to VNFs of SFC are available, it is considered that the current SFC is available. In simulating the faults of underlying network nodes, two strategies are adopted: random simulation faults and resource feature simulation faults. Random simulation of faults refers to randomly selecting [2%, 3%] of the underlying nodes as faulty nodes. Simulating faults based on resource characteristics refers to arranging the underlying nodes in descending order of resource utilisation, and selecting the underlying nodes with higher resource utilisation [2%, 3%] as faulty nodes.

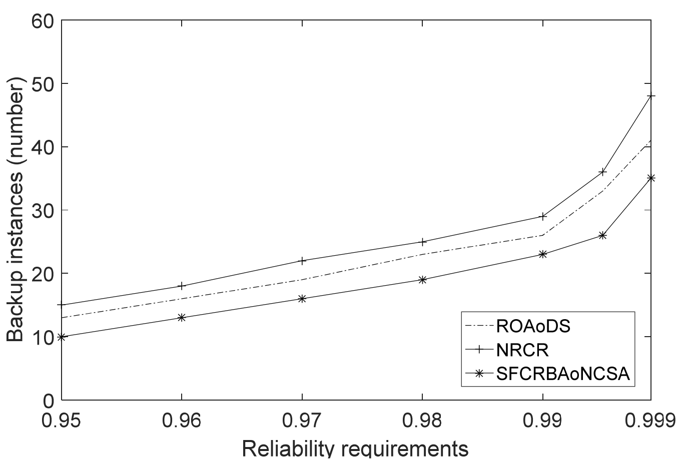
In terms of the acceptance rate of SFC, the underlying network gains revenue by carrying SFC, so whether the underlying network can carry more SFC is an important evaluation indicator. This section uses the acceptance rate of SFC to analyse the impact of backup resources on network performance. The acceptance rate of SFC refers to the success rate of the underlying network allocating resources to SFC after resource backup. This indicator can analyse the impact of backup strategies on the utilisation of underlying network resources.

5.2 Backup resource consumption

The experimental results of the number of backup instances and backup bandwidth resource consumption are shown in Figure 1 and Figure 2. The X axis in the figure shows that the reliability requirement of SFC has increased from 0.95 to 0.999, which is used to simulate the environment with different user requirements.

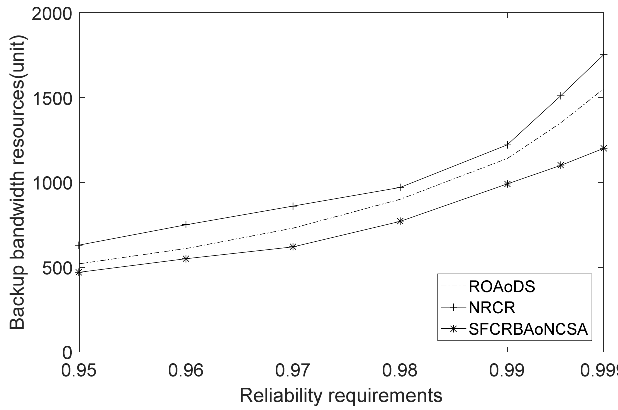
It can be seen from the comparison results of the number of backup instances in Figure 1 that the number of backup instances under the three algorithms is increasing with the improvement of reliability requirements. Especially when the SFC reliability requirement is greater than 0.99, the number of backup instances increases rapidly. In terms of performance analysis of the three algorithms, NRCR consumes the most backup instances, followed by ROAODS. When ROAODS selects the underlying node, it gives priority to the underlying node resources with high backup advantages, thus quickly improving the reliability of SFC. The algorithm NRCR only considers meeting the resource requirements when selecting the bottom node. Therefore, the ROAODS algorithm requires fewer instances. The algorithm ROAODS consumes more backup instances than the algorithm SFCRBAONCSA. Because the algorithm ROAODS does not consider the sharing of backup resources. Because backup resource sharing can select resources that meet multiple VNF instances at the same time, fewer backup instances are required.

Figure 1 Comparison of backup instances



It can be seen from the comparison results of backup bandwidth resource consumption in Figure 2 that with the increase of reliability requirements, the bandwidth resources consumed by the three algorithms are gradually increasing. Especially when the reliability requirement is greater than 0.99, the bandwidth requirements of the three algorithms increase rapidly. Compared with the three algorithms, ROAODS algorithm consumes less backup bandwidth resources, while NRCR algorithm consumes the most backup bandwidth resources. When ROAODS selects the bottom node, it gives priority to the bottom node resources with high backup advantages. These nodes have more adjacent nodes and links, which makes it easier to find the links connected to the adjacent nodes of the bottom node to be backed up, thus reducing the consumption of link resources. The backup bandwidth resource consumption of algorithm ROAODS is more than that of algorithm SFCRBAONCSA. The algorithm SFCRBAONCSA selects the nearest underlying node that can be shared by multiple VNF underlying nodes to be backed up as the backup node resource, requiring less backup link resources. Therefore, the algorithm SFCRBAONCSA in this paper consumes the least backup bandwidth resources.

Figure 2 Comparison of backup bandwidth resource consumption



5.3 *Availability of SFC*

Figure 3 shows the comparison results of SFC availability when randomly simulating faults. The X axis indicates that the number of underlying nodes has increased from 50 to 100, which is used to simulate network environments of different scales. It can be seen from the figure that under different network sizes, the availability of SFC under the three algorithms fluctuates slightly. It shows that different network scale environments have little impact on the availability of the three algorithms SFC. In terms of performance comparison of the three algorithms, the SFC availability results of the three algorithms are not different, indicating that the three algorithms have good convergence results. The availability of SFC of ROAODS algorithm is slightly better than that of NRCR algorithm. Although the failures are random, when ROAODS selects the underlying node, it gives priority to the resources with high backup resource advantages, so as to quickly ensure the high reliability of SFC routing. The availability of SFC of algorithm SFCRBAONCSA is slightly better than that of algorithm ROAODS. Because the algorithm SFCRBAONCSA considers the sharing relationship of backup resources, so as to improve the availability.

Figure 4 shows the comparison results of SFC availability when simulating faults according to resource characteristics. It can be seen from the figure that with the increase of the number of underlying nodes, the availability of SFC under the three algorithms tends to converge. This shows that the size of the underlying network has little impact on the availability of SFC. The availability of SFC of algorithm NRCR is significantly lower than that of algorithm ROAODS and algorithm SFCRBAONCSA. When the algorithm ROAODS and the algorithm SFCRBAONCSA select the backup bottom node, the selected bottom node is the bottom node with greater backup advantages. The utilisation rate of these nodes is low, so the reliability is high. When the resource with high utilisation fails, the reliability of the SFC in this paper is less affected. Compared with the algorithm ROAODS, the algorithm SFCRBAONCSA further optimises the sharing relationship of SFC backup resources and improves the reliability of SFC links.

Figure 3 Comparison of SFC availability under random simulated faults

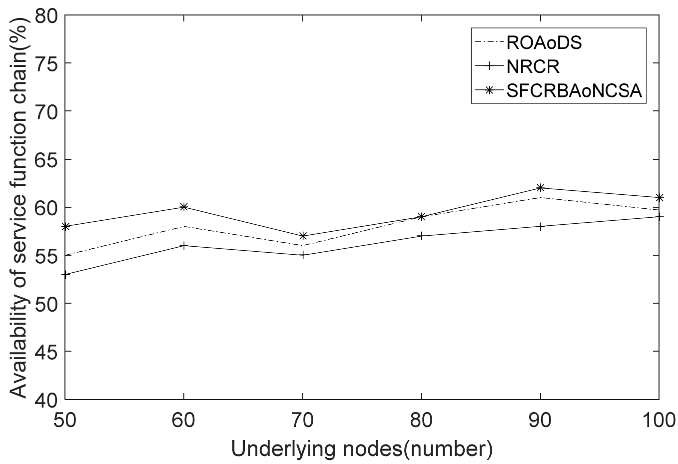
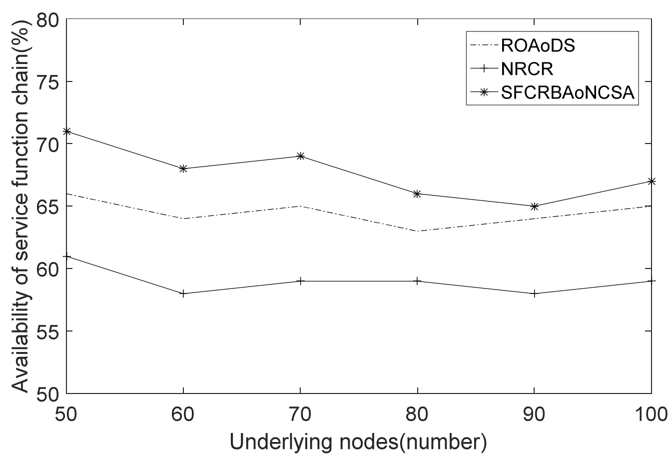


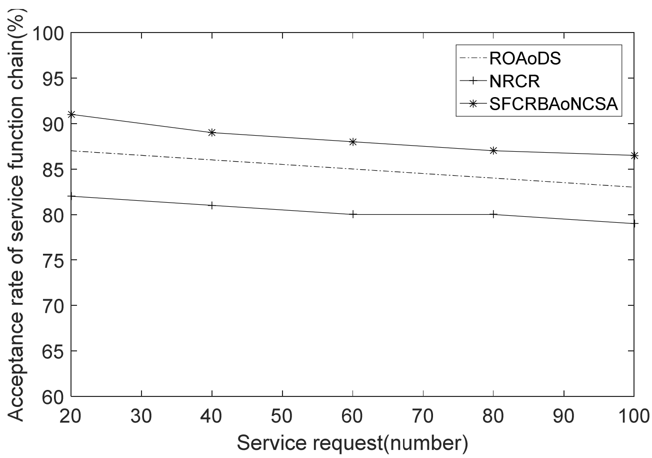
Figure 4 Comparison of SFC availability when failures are related to resource characteristics



5.4 Acceptance rate of SFC

Figure 5 shows the acceptance rate comparison results of SFCs. The X axis indicates that the number of service requests has increased from 20 to 100. It can be seen from the figure that with the increase of the number of service requests, the acceptance rate of SFC under the three algorithms is decreasing. Especially when there are more than 60 service requests, the acceptance rate of SFC decreases rapidly. This shows that when the number of SFCs is greater than 60, the resources required by SFCs can no longer be met by the underlying network resources. In terms of performance analysis of the three algorithms, the SFC acceptance rate of ROAODS algorithm is slightly higher than that of NRCR algorithm. Because the selected backup node has greater centrality, it can realise the connection between the backup node and the adjacent node of the node to be backed up through fewer links, saving more underlying link resources. The algorithm NRCR may require more link resources in order to select resources with higher reliability. The SFC acceptance rate of algorithm ROAODS is lower than that of algorithm SFCRBAONCSA. The algorithm SFCRBAONCSA considers the sharing relationship of backup resources based on the algorithm ROAODS, uses fewer backup resources, provides more available resources for accepting SFC, and improves the acceptance rate of SFC.

Figure 5 Acceptance rate comparison of SFC



By comparing the algorithm in this paper with the classical algorithm in current research, we can see that the algorithm in this paper saves the resource consumption of backup instances and backup bandwidth, thus further improving the availability and acceptance rate of SFC. Therefore, the SFCRBAONCSA method proposed in this paper has superior performance.

6 Conclusions

The research purpose of this article is to design an algorithm that minimises resource overhead while ensuring the reliability requirements of network services. To address this issue, this article designs a SFC resource backup algorithm based on network features and sharing advantages. The simulation results show that the algorithm in this paper saves resource consumption for backup instances and backup bandwidth, and improves the availability and acceptance rate of SFC. Compared to existing research, the research results of this article have superior performance. The research in this article found that by comprehensively evaluating the superiority of VNF instances to be optimised and optional backup underlying nodes to determine the priority of resource backup, the shared characteristics of network resources can be fully utilised to optimise resource deployment. Therefore, the algorithm in this article formulates SFC reliability optimisation deployment strategies based on the network characteristics of the underlying nodes and VNF instances, which can minimise resource overhead while ensuring the reliability requirements of network services, achieving the research purpose of this article.

For the reliability of the network, further research is needed from the following two dimensions. Firstly, it is not only necessary to ensure network reliability to meet business needs during resource allocation, but also to ensure network reliability during network operation. Secondly, it is not only necessary to ensure the reliability of network resources from the perspective of resource sharing, but also to consider how to implement resource sharing under the premise of business security. Therefore, in the next step of work, based on the research results of this article, further research will be conducted on network reliability strategies during network operation and resource sharing strategies under the premise of business security.

References

Alvarez-Hamelin, J., Dall'Asta, L., Barrat, A. et al. (2005) 'Large scale networks fingerprinting and visualization using the k-core decomposition', *Advances in Neural Information Processing Systems*, Vol. 18, No. 1, p.41.

Anthoniraj, S. and Saraswathi, S. (2018) 'Frequent failure monitoring and reporting in virtualisation environment using backing algorithm technique', *International Journal of Information and Communication Technology*, Vol. 12, Nos. 1–2, pp.131–142.

Bangqiang, L., Zhoumo, Z. and Jing, D. (2016) 'A new method of fault tolerance based on partial reconfiguration', *International Journal of Information and Communication Technology*, Vol. 9, No. 2, pp.232–241.

Cai, S., Zhou, F., Zhang, Z. et al. (2021) 'Disaster-resilient service function chain embedding based on multi-path routing', *IEEE INFOCOM 2021-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, IEEE, pp.1–7.

Cotroneo, D., De Simone, L., Iannillo, A.K. et al. (2014) 'Network function virtualization: Challenges and directions for reliability assurance', *2014 IEEE International Symposium on Software Reliability Engineering Workshops*, IEEE, pp.37–42.

Dorogovtsev, S.N., Goltsev, A.V. and Mendes, J.F.F. (2006) 'K-core organization of complex networks', *Physical Review Letters*, Vol. 96, No. 4, p.40601.

GT-ITM (2000) *Modeling Topology of Large Internetworks* [online] http://www.cc.gatech.edu/projects/gtitm/.

Guo, S., Qi, Y., Jin, Y. et al. (2021) 'Endogenous trusted DRL-based service function chain orchestration for IoT', *IEEE Transactions on Computers*, Vol. 71, No. 2, pp.397–406.

He, F., Sato, T. and Oki, E. (2019) 'Backup resource allocation model for virtual networks with probabilistic protection against multiple facility node failures', *2019 15th International Conference on the Design of Reliable Communication Networks (DRCN)*, IEEE, pp.37–42.

Kang, R., He, F. and Oki, E. (2021) 'Virtual network function allocation in service function chains using backups with availability schedule', *IEEE Transactions on Network and Service Management*, Vol. 18, No. 4, pp.4294–4310.

Latchoumy, P. and Khader, P.S.A. (2015) 'Reliable job execution with process failure recovery in computational grid', *International Journal of Information and Communication Technology*, Vol. 7, No. 6, pp.607–631.

Liu, H., Ding, S., Wang, S. et al. (2022) 'Multi-objective optimization service function chain placement algorithm based on reinforcement learning', *Journal of Network and Systems Management*, Vol. 30, No. 4, pp.1–25.

Magoula, L., Barmpounakis, S., Stavrakakis, I. et al. (2021) 'A genetic algorithm approach for service function chain placement in 5G and beyond, virtualized edge networks', *Computer Networks*, Vol. 195, No. 9, pp.1–12.

Qing, S., Liao, J., Wang, J. et al. (2012) 'Hybrid virtual network embedding with K-core decomposition and time-oriented priority', in *Proceedings of IEEE International Conference on Communications (ICC 2012)*, Ottawa, Canada, pp.2695–2699.

Qu, L., Assi, C., Shaban, K. et al. (2017) 'A reliability-aware network service chain provisioning with delay guarantees in NFV-enabled enterprise datacenter networks', *IEEE Transactions on Network and Service Management*, Vol. 14, No. 3, pp.554–568.

Sato, T., He, F., Oki, E. et al. (2018) 'Implementation and testing of failure recovery based on backup resource sharing model for distributed cloud computing system', *2018 IEEE 7th international conference on cloud networking (CloudNet)*, IEEE, pp.1–3.

Soualah, O., Mechtri, M., Ghribi, C. et al. (2017) 'A link failure recovery algorithm for virtual network function chaining', *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, IEEE, pp.213–221.

Sun, J., Zhu, G., Sun, G. et al. (2018) 'A reliability-aware approach for resource efficient virtual network function deployment', *IEEE Access*, Vol. 6, No. 3, pp.18238–18250.

Wang, Y., Zhang, L., Yu, P. et al. (2021) 'Reliability-oriented and resource-efficient service function chain construction and backup', *IEEE Transactions on Network and Service Management*, Vol. 18, No. 1, pp.240–257.

Yang, Y., Meng, X.R., Kang, Q.Y. et al. (2021) 'Reliable service function chain deployment method based on traffic optimization', *Systems Engineering and Electronics*, Vol. 43, No. 10, pp.3017–3025.

Zegura, E.W., Calvert, K.L. and Bhattacharjee, S. (1996) 'How to model an internetwork', *Proceedings of IEEE INFOCOM'96. Conference on Computer Communications*, IEEE, Vol. 2, pp.594–602.

Zhai, D., Meng, X.R., Kang, Q.Y. et al. (2020) 'Service function chain deployment method for delay and reliability optimization', *Journal of Electronics and Information*, Vol. 42, No. 10, pp.2386–2393.

Zhang, L., Wang, Y., Qiu, X. et al. (2019) 'Redundancy mechanism of service function chain with node-ranking algorithm', *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, IEEE, pp.586–589.

Zhu, M., He, F. and Oki, E. (2020) 'Multiple backup resource allocation with workload-dependent failure probability', *GLOBECOM 2020-2020 IEEE Global Communications Conference*, IEEE, pp.1–6.

Zhu, Y., Yao, H., Mai, T. et al. (2022) 'Multi-agent reinforcement learning aided service function chain deployment for internet of things', *IEEE Internet of Things Journal*, Vol. 9, No. 17, pp.15674–15684.