Studies on Cost-Effective Service Chain Construction with VNF Sharing Model and Security-Level Management

A Dissertation Submitted to the University of Fukui
for Degree of Doctor of Engineering

Studies on Cost-Effective Service Chain Construction with VNF Sharing Model and Security-Level Management

March 2024

Daisuke Amaya

福井大学
学位審査論文 [博士 (工学)]

VNF 共有モデルとセキュリティレベルマネジメントを用いた費用対効果の高いサービスチェイン構築に関する研究

2024 年 3 月

天谷 大輔

A Dissertation
submitted to Graduate School of Engineering,
University of Fukui
in partial fulfillment of the requirements for the degree of
DOCTOR of ENGINEERING

Daisuke Amaya

Dissertation Committee:

Professor Takuji Tachibana	(Supervisor)
Professor Shin-ichiro Mori	(Co-supervisor)
Professor Ken Higuchi	(Co-supervisor)

Abstract

In the current society, communication networks are indispensable infrastructure for the transmission of information and service delivery. In traditional communication networks, dedicated physical hardware resources are deployed for each network function. The introduction of new network services or the upgrade of existing networks is a time-consuming and costly endeavor for network operators. However, with the rapid digitization in recent years, the requirements for communication networks will become increasingly diverse and complex. To address this new communication environment, network virtualization technology has emerged. This technology is gaining attention as a networking paradigm that can replace traditional communication networks and provide flexibility and agility to communication networks.

Network virtualization involves creating a set of sliced physical network resources, such as physical links and routers, which form a virtual network capable of running its own protocols, routing processes, and services. In recent times, the European Telecommunications Standards Institute (ETSI) has developed network functions virtualization (NFV). By using NFV technology, various network functions like deep packet inspection (DPI), firewall, routers, network address translation (NAT), and so on can be implemented as virtual machines on commercial servers. These network functions are referred to as virtual network functions (VNFs). Moreover, service chaining has garnered attention as an emerging technology that allows for the dynamic and flexible provision of network services by connecting VNFs in the right order. Leveraging this technology, network operators can offer a variety of network services to their users. In cases where multiple service chains utilize the same network function on the same node, these service chains can share the VNF. Actively sharing VNFs can reduce the overall resource consumption in the network since deploying VNFs consumes resources like CPU and memory on the node. Therefore, VNF sharing can also lead to reductions in capital expenditure and operational expense. However, VNF sharing brings forth several issues. Firstly, there is the issue of VNF processing performance. If there are VNFs with insufficient resources within a service chain, packets transmitted across multiple service chains may fail to meet quality of service (QoS) requirements. It is important to carefully select VNF instances shared among

multiple service chains and determine the resource allocation for these selected VNFs. Secondly, there are cybersecurity concerns. VNFs themselves may become the source or target of cyberattacks. Malicious VNFs can attack the node where they are installed, potentially compromising other VNFs on the same node. To safeguard against cyberattacks, each service chain demands different security requirements such as traffic separation, protection against cyberattacks, and authenticity and integrity checks.

In this dissertation, we propose two heuristic-based approaches for service chain construction. Firstly, we model VNFs as M/M/1/K queueing models to evaluate the relationship between resource allocation and loss probability, offering a cost-effective service chain construction proposal. The method makes decisions regarding VNF sharing, VNF placement, amount of resources for each VNF, and the transmission route of each service chain to minimize the cost of service chain construction. Secondly, we introduce the concept of security levels into the first model and propose service chain construction using security level management. This approach enhances the security levels of VNFs and nodes through security management (enhancing security mechanisms). However, as enhancing security mechanisms increases costs, we also consider cost-effectiveness in service chain construction using this method. Thus, we formulate cost optimization problems for each service chain construction and propose heuristic algorithms to solve these optimization problems.

We evaluate the performance of the proposed methods through simulations in various scenarios to investigate their effectiveness. The first numerical examples show that our proposed method can construct service chains that minimize the cost for VNFs regardless of the number of service chains, the requirement in terms of the loss probability, the amount of traffic for a service chain, the amount of resources for a node, and the network topologies. In particular, we find that our proposed method can minimize the cost for the VNFs by sharing not only the VNF instances but also the resources for processing among multiple service chains. Moreover, the second numerical examples demonstrate that our proposed method can construct service chaining to minimize the total security and resource consumption costs. We find that our proposed method can significantly decrease the total cost regardless

of the three network topologies. Additionally, the calculation time for the proposed method is significantly smaller compared with the calculation time for solving the optimization problem.

Key words:

Network function virtualization, virtual network function, service chaining, queueing model, security level, optimization problem, heuristic algorithm

概 要

現在の社会において，通信ネットワークは情報の伝送やサービス提供のために不可欠な基盤である．従来の通信ネットワークでは，各ネットワーク機能に対して専用の物理ハードウェアリソースが展開されているため，新しいネットワークサービスの導入や既存のネットワークのアップグレードは，ネットワークオペレータにとって手間と費用のかかる作業である．しかし，近年の急速なデジタル化により，通信ネットワークの要件はますます多様かつ複雑になっている．この新しい通信環境に対処するために，ネットワーク仮想化技術が登場している．この技術は，従来の通信ネットワークを代替し，通信ネットワークに柔軟性と俊敏性を提供できるネットワーキングパラダイムとして注目されている．

ネットワーク仮想化には，物理リンクやルータなどのスライスされた物理ネットワークリソースのセットを作成し，独自のプロトコル，ルーティングプロセス，およびサービスを実行できる仮想ネットワークが含まれる．近年，欧州電気通信標準化協会（ETSI）はネットワーク機能仮想化（NFV）を開発した．NFV 技術を使用することで，Deep Packet Inspection（DPI），ファイアウォール，ルータ，ネットワークアドレス変換（NAT）など，さまざまなネットワーク機能を商用サーバ上の仮想マシンとして実装できる．これらのネットワーク機能は仮想ネットワーク機能（VNF）と呼ばれ，さらに，サービスチェイニングは VNF を適切な順序で接続することにより，ネットワークサービスの動的で柔軟な提供を可能にする新興の技術として注目されている．この技術を活用することで，ネットワークオペレータはさまざまなネットワークサービスをユーザーに提供できる．同じノード上で複数のサービスチェインが同じネットワーク機能を利用する場合，これらのサービスチェインは VNF を共有可能である．VNF インスタンスを展開することはノード上の CPU やメモリなどのリソースを消費するため，積極的に VNF を共有することで，ネットワーク全体のリソース消費を削減できる．したがって，VNF の共有は資本支出と運用費用の削減にもつながる．しかし，VNF の共有にはいくつかの課題が存在する．まず第一に，サービスチェイン内のいくつかの VNF に十分なリソースがない場合，複数のサービスチェインを介して転送されるパケットはサービス品質（QoS）の要件を満たさない可能性がある．複数のサービスチェイン間で共有される VNF インスタンスを慎重に選択し，これらの選択された VNF に対するリソースの割り当てを決定するこ

とが重要である．第二に，サイバーセキュリティに関する懸念が存在する．VNF 自体がサイバー攻撃のソースまたはターゲットになる可能性があり，悪意のある VNF はそれらがインストールされているノードを攻撃し，同じノード上の他の VNF を危険にさらす可能性がある．サイバー攻撃から保護するために，各サービスチェインはトラフィックの分離，サイバー攻撃に対する保護，および信頼性と整合性の確認など，異なるセキュリティ要件を要求する．

この論文では，サービスチェインの構築に向けた 2 つのヒューリスティクスベースのアプローチを提案する．まず第一に，VNF を M/M/1/K キューイングモデルとしてモデル化し，リソース割り当てと損失確率の関係を評価して，費用対効果のあるサービスチェインの構築を提案する．この方法は，VNF の共有，VNF の配置，各 VNF のリソース量，および各サービスチェインの伝送経路に関する決定を行い，サービスチェイン構築のコストを最小化する．次に，第一のモデルにセキュリティレベルの概念を導入し，セキュリティレベル管理を使用したサービスチェインの構築を提案する．このアプローチは，セキュリティ管理を通じて VNF とノードのセキュリティレベルを向上させる．ただし，セキュリティメカニズムの向上はコストを増加させるため，この方法を使用したサービスチェインの構築においても費用対効果を考慮する．したがって，各サービスチェインの構築に対するコスト最適化の問題を定式化し，これらの最適化問題を解決するためのヒューリスティックアルゴリズムを提案する．

提案手法の性能をさまざまなシナリオでのシミュレーションを通じて評価し，その有効性を調査する．最初の数値例は，提案手法がサービスチェインを構築でき，サービスチェインの数，損失確率の要件，サービスチェインのトラヒック量，ノードのリソース量，およびネットワークのトポロジに関係なく，VNF のコストを最小化できることを示している．特に，提案手法が VNF のインスタンスだけでなく，処理リソースも複数のサービスチェイン間で共有することによって，VNF のコストを最小化できることがわかる．さらに，2 番目の数値例は，提案手法がセキュリティおよびリソース消費の総コストを最小化するためのサービスチェインを構築できることを示している．提案手法が 3 つのネットワークトポロジに関係なく，総コストを著しく減少させることができることがわかる．さらに，提案手法の計算時間は，最適化問題を解決するための計算時間と比較して著しく短縮されている．

キーワード：
　ネットワーク機能仮想化，仮想ネットワーク機能，サービスチェイニング，待ち行列モデル，セキュリティレベル，最適化問題，ヒューリスティックアルゴリズム

研究業績

- 学術論文

  [1] ◯D. Amaya and T. Tachibana, "Heuristic-based Service Chain Construction with Security-Level Management," *IEICE TRANSACTIONS on Communications*, vol. E106-B, no. 12, pp. 1380–1391, Dec. 2023.

  [2] ◯D. Amaya and T. Tachibana, "Cost-Effective Service Chain Construction with VNF Sharing Model Based on Finite Capacity Queue," *IEICE TRANSACTIONS on Communications*, vol. E105-B, no. 11, pp. 1361–1371, Nov. 2022.

- 国際会議（査読付）

  [1] ◯ D. Amaya and T. Tachibana, "Security Level Management of Physical Servers Based on Optimization Problem for Cost-Effective Service Chain Construction," in *Proc. IEICE International Conference on Emerging Technologies for Communications*, Nov. 2022.

- 研究会

  [1] 天谷 大輔, 橘 拓至, "待ち行列理論に基づくVNF共有モデルを利用した最適サービスチェイン構成法," 超知性ネットワーキングに関する分野横断型研究会, Nov. 2021.

Acknowledgements

I would like to offer my special thanks to Professor Takuji Tachibana of the University of Fukui for his careful guidance and appropriate assistance. He gave me much appropriate advice about studies and taught me how to write the dissertation carefully. Without his encouragement, guidance, and support, this dissertation would not have materialized.

I am deeply grateful to Professor Shin-ichiro Mori of the University of Fukui. He reviewed my dissertation, and he gave insightful comments and suggestions.

I express my appreciation to Professor Ken Higuchi of the University of Fukui. He reviewed my dissertation, and he gave me constructive comments and warm encouragement.

I am deeply grateful to Associate Professor Tomoya Kawakami of University of Fukui. He taught me many helpful advices and comments for my study in his seminar.

In addition, I would like to thank all members of Tachibana's laboratory. Finally, for their moral support and warm encouragement, I would also like to express my gratitude to my family.

Contents

List of Figures

Chapter 1

Introduction

1.1 Network Functions Virtualization

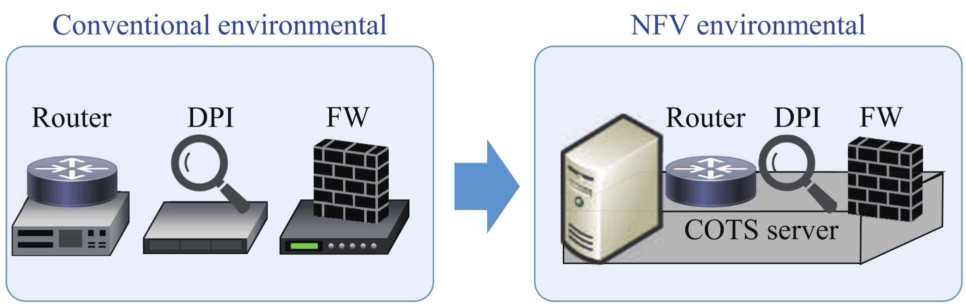

Figure 1.1: Network Function Virtualization

In the current society, communication networks are indispensable infrastructure for the transmission of information and service delivery. Network operators provide network services using various network devices such as routers, switches, and firewalls [1]. In traditional communication networks, dedicated physical hardware resources are deployed for each network function [2]. Consequently, the introduction of new network services or the upgrade of existing networks is a time-consuming and costly endeavor for network operators [3, 4]. However, with the rapid digitization in recent years, the requirements for communication networks will become increasingly diverse and complex [5, 6]. To address this new communication environment, network virtualization technology has emerged. This technology is gaining attention as a networking paradigm that can replace traditional communication networks and provide flexibility and agility to communication networks [7].

Network virtualization involves creating a set of sliced physical network resources, such as physical links and routers, which form a virtual network capable of running its own protocols, routing processes, and services [8]. In other words, network virtualization enables the construction of virtual networks that can meet various service requirements [9]. Furthermore, in recent times, the European Telecommunications Standards Institute (ETSI) has developed network functions virtualization (NFV). This concept in network architecture leverages virtualization technology to migrate network functions from dedicated physical hardware to software applications [10]. As shown in Fig. 1.1, by using NFV technology, various network functions like deep packet inspection (DPI), firewall, routers, network address translation (NAT), and so on can be implemented as virtual machines on commercial servers [11, 12]. These network functions are referred to as virtual network functions (VNFs). In environments where NFV is available, dedicated network equipment for providing network functions is not necessary, leading to significant capital expenditure reductions. In addition, VNFs can be easily installed or removed from any commercial off-the-shelf (COTS) server, thereby reducing operational expense [13, 14].

NFV is a valuable technology for network operators, but the correct flow of packets requires the proper interconnection of VNFs [15]. Service chaining has garnered attention as an emerging technology that allows for the dynamic and flexible provision of network services by connecting VNFs in the right order [16]. Leveraging this technology, network operators can offer a variety of network services to their users. In cases where multiple service chains utilize the same network function on the same node, these service chains can share the VNF. Actively sharing VNFs can reduce the overall resource consumption in the network since deploying VNFs consumes resources like CPU and memory on the node [17]. Therefore, VNF sharing can also lead to reductions in capital expenditure and operational expense. However, VNF sharing brings forth several issues. Firstly, there is the issue of VNF processing performance. If there are VNFs with insufficient resources within a service chain, packets transmitted across multiple service chains may fail to meet quality of service

(QoS) requirements [18, 19, 20, 21]. Secondly, there are cybersecurity concerns. Sharing the same VNF among service chains with different security requirements or co-locating VNFs with varying security requirements on the same node may give rise to security issues [22].

1.2 Issues in Service Chaining

Each service chain must meet QoS requirements such as packet loss probability [23, 24, 25, 26]. The number of packets a VNF can process per unit time depends on the processing resource capacity of the VNF [16, 26, 27]. When a service chain includes VNFs with low processing resources, the packets transmitted across the service chain may fail to meet QoS requirements [18, 19, 20, 21]. To enhance QoS, it is necessary to increase the processing resource capacity of VNFs. However, an increase in VNF processing resource capacity leads to elevated costs related to service chain management, energy consumption, and other factors [30, 31, 32]. Here, VNF resources are classified into placement and processing resources, including CPU and memory. Placement resources are required for creating a VNF instance on a server and encompass resources consumed by tasks such as image processing and boot processing. The amount of placement resources remains constant and independent of the processing performance of the VNF, and it increases the total amount of placement resources when more VNFs are deployed. By contrast, when VNFs are shared among multiple service chains, the number of VNFs does not increase, allowing for a reduction in the amount of placement resources for VNFs. However, as the number of packets processed within the VNF increases, there is a corresponding need to augment the processing resource capacity. As a result, the cost of service chain construction is significantly influenced by VNF sharing. Therefore, it is important to carefully select VNF instances shared among multiple service chains and determine the resource allocation for these selected VNFs.

Cyberattacks, including unauthorized access and malware, have seen an increase in both frequency and complexity in recent years[33, 34, 35, 36]. Attackers attempt to eavesdrop on or alter packets passing between VNFs and NFV Management and Orchestration (MANO), as well as packets within

NFV MANO itself [35]. Moreover, NFV MANO plays a critical role in managing the resources utilized by VNFs. An attack on NFV MANO has the potential to result in severe consequences, including data theft and service denial, among other adverse outcomes. Encryption and monitoring have proven effective in preventing such eavesdropping and data theft. Additionally, VNFs themselves may become the source or target of cyberattacks [35]. Since VNFs are often provided by vendors other than the infrastructure provider, VNFs may have security vulnerabilities or even be designed as malware for carrying out attacks [22]. If the node where a VNF is installed is compromised, the VNF can be easily exploited due to its security vulnerabilities. Conversely, malicious VNFs can attack the node where they are installed, potentially compromising other VNFs on the same node [38]. To mitigate or prevent these cyberattacks, nodes and VNFs rely on each other for appropriate security mechanisms, including packet filtering, access control, monitoring, and antivirus mechanisms [39]. RFC 7665, standardized by the Internet Engineering Task Force, provides additional security considerations for service chaining [40]. If an untrusted sender can inject packets that are treated as properly classified for service chaining, attackers can disrupt all users and services for each service chain by conducting various types of cyberattacks. To safeguard against cyberattacks, each service chain demands different security requirements such as traffic separation, protection against cyberattacks, and authenticity and integrity checks.

The technologies of NFV and service chaining offer network operators a range of benefits and issues. VNF placement and resource allocation require careful consideration and have been extensively studied [23, 24, 25, 26, 27, 28, 29]. [23, 24, 25] focused on minimizing the construction cost of service chains while meeting latency requirements. These papers primarily utilized infinite-capacity queuing models to investigate variations in latency. However, in environments characterized by the transmission of a large number of packets or limited resource availability, there arises a shortage of VNF resources required for service chain construction. Therefore, effectively utilizing overall resources becomes imperative. Moreover, in such environments, it becomes necessary to fulfill QoS requirements from the perspective of packet loss probability for each service chain. However, it is worth noting that research on considering packet loss probability in the context of service chaining is scarce.

In addition, there has been limited research on security aspects of service chaining. The concept of security levels, which represents the effectiveness of security mechanisms in preventing cyberattacks, was introduced for the first time in the context of network virtualization and was used to evaluate the effectiveness of different security mechanisms [22, 37]. Following the approach of [22, 37], security levels were introduced for VNFs and nodes, and a method for constructing service chains that meet security requirements was proposed [41]. However, this method does not intend to manage or control security levels, and as a result, some service chains may not be constructed due to insufficient security levels.

In this dissertation, we propose two heuristic-based approaches for solving optimization problems related to service chain construction within practical time. The service chain construction problem has already been proven to be NP-hard, and most algorithms addressing it were designed with heuristics to consider the complexity of the problem and computation time [17]. Although better near-optimal solutions may be obtainable through approximation algorithms such as linear approximations, it is difficult to solve large-scale problems within practical time. Additionally, we assume that heuristic algorithms, which are easy to modify and can consistently yield the same solution based on strategy, are preferable from a network operation perspective. Firstly, we model VNFs as M/M/1/K queueing models to evaluate the relationship between resource allocation and loss probability, offering a cost-effective service chain construction proposal. Secondly, we introduce the concept of security levels into this model and propose service chain construction using security level management. This approach enhances the security levels of VNFs and nodes through security management (enhancing security mechanisms). However, as enhancing security mechanisms increases costs, we also consider cost-effectiveness in service chain construction using this method. Thus, we formulate cost optimization problems for each service chain construction and propose heuristic algorithms to solve these optimization problems. We evaluate the performance of the proposed methods through simulations in various scenarios to investigate their effectiveness.

1.3 Structure of This Dissertation

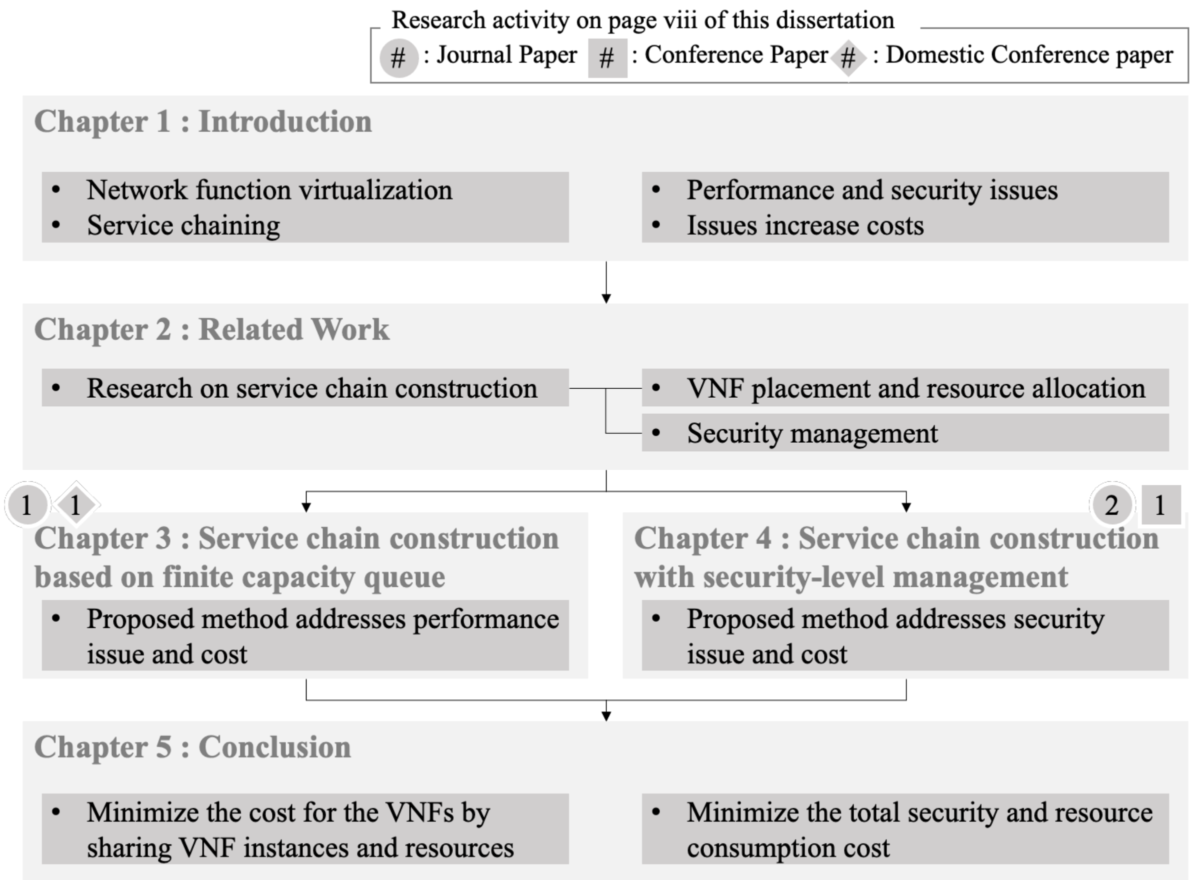

Figure 1.2: Structure of This Dissertation

The rest of this dissertation is organized as shown in Fig. 1.2. Chapter 2 describes the related work on service chain construction with a focus on the placement and resource allocation of VNFs, as well as security management with security level and demand. Chapter 3 proposes a service chain construction with a VNF sharing model based on a finite capacity queue, an aspect not primarily addressed in related work. It tackles issues related to processing resources and cost, as mentioned in Section 1.2. Additionally, Chapter 4 describes the proposed service chain construction with security-level management, addressing both security and cost-related concerns, as discussed in Section 1.2. Finally, Chapter 5 denotes conclusions.

Chapter 2

Related Work

In this chapter, we describe related work on service chain construction and security management with security level and demand.

2.1 VNF Placement and Resource Allocation for Service Chain Construction

In this section, we explain related work on the VNF placement and the resource allocation for service chaining construction. Here, [26] and [28] dealt with the problem of traffic load on service chains but did not consider the loss probability. Moreover, [23] modeled VNFs as M/M/1 queue and focused on the latency requirements. However, we model VNFs as M/M/1/K queue and deal with the amount of traffic which takes into account the loss probability.

[26] formulated an optimization problem to maximize total data throughput in resource-constrained NFV environments. They proposed a heuristic service chain construction algorithm to determine the placement of VNFs, resource allocation, and transmission paths for this optimization problem. This paper classified VNF resource requirements into constant placement resources and variable processing resources, which vary proportionally with traffic volume, based on [42]. Data throughput was defined as the amount of traffic that a VNF can process with its variable resource allocation. Therefore, when sharing VNFs across multiple service chains, the consumption of placement resources can be minimized. This allows for efficient allocation of resources

across the entire network, maximizing the total data throughput. However, this data throughput is solely determined by the amount of processing resources and does not consider packet loss probabilities.

[28] conducted research on optimal service chain construction based on heuristic algorithms with the aim of reducing the maximum traffic load on links and the total number of VNFs. This heuristic algorithm reduces the number of VNF instances by sharing VNFs among multiple service chains. Moreover, by appropriately controlling the order in which packets traverse VNFs, it reduces the maximum traffic load on links. Since the objective function includes parameters, changes in these parameters can prioritize the reduction of either the maximum traffic load or the total number of VNFs.

In [23], the authors modeled VNFs as M/M/1 queues and defined the arrival rate and service rate for VNFs based on the number of service chains utilizing those VNFs. The arrival rate represents the number of service chains utilizing a particular VNF, while the service rate corresponds to the resource allocation required for processing the service chains. To minimize costs, they formulated an optimization problem to determine VNF placement, resource allocation, arrival rates, and service rates. Since the queuing model is considered to have an infinite capacity, this paper focuses on constructing service chains to meet the latency requirements of each VNF rather than calculating packet loss probabilities.

2.2 Security Management with Security Level and Demand

In this section, we provide an overview of prior research focusing on security management involving security levels and demand. Here, [22] and [37] addressed the challenge of embedding virtual networks into physical networks, and [41], [43], [44], and [45] dealt with the issues related to service chain construction. [22], [37], [41], [43], and [44] formulated the optimization problems and proposed algorithms to obtain the optimal solutions. However, these problems did not take into account the augmentation of security levels to construct all service chains.

[22] introduced the concept of security levels to signify standard protection

measures. Each physical and virtual node is assigned a specific security level, which can be determined by the network operator or the user. A physical node with a higher security level incorporates more advanced security mechanisms when hosting virtual nodes. For instance, a physical node equipped with data encryption has a higher security level compared to one without encryption capabilities. If the security level of the physical node is equal to or greater than that of the virtual node, the virtual node can be embedded within the physical node.

[41] extended the notion of security levels introduced in [22] to be employed in service chaining. Each physical node and VNF is assigned a security level, which the network provider can select based on user requirements. Users must specify a simple security quality level, such as high, medium, or low, for particular network services they require. In this approach, VNFs are deployed to meet the security level of each service chain. Moreover, more service chains are constructed to obtain higher revenue and lower cost.

[43] proposed an optimization model for security and delay in service chain construction. This model incorporates security levels and demands for VNFs and nodes, similar to [41]. The proposed method maximizes the ratio of revenue to cost while ensuring security requirements are met. Revenue is influenced by both security demands and the resources allocated to service chains, while costs are influenced by the number of hops, in addition to them. Optimizing the transmission delay involves reducing the number of hops to maximize the revenue-to-cost ratio. Additionally, [44] introduced a security-aware service chaining approach to optimize load balancing and transmission delay. In this method, the revenue-to-cost ratio is calculated by considering the set of successfully constructed service chains during each time period. To achieve this objective, the method takes into account bottleneck nodes and links as constraints.

[45] proposed security orchestration in the SDN/NFV environment to ensure both security and optimal service chaining. This model introduces security levels for communication links. Communication links within the same cloud are considered more secure, whereas links between users and interclouds that lack security mechanisms, such as IPSEC, SSL, and datagram transport layer security, are assigned lower security levels. Additionally, the

study stipulates that all communication links interconnecting VNFs within the same service chains must have security levels higher than the standard security level.

Chapter 3

Cost-Effective Service Chain Construction with VNF Sharing Model Based on Finite Capacity Queue

In this chapter, we propose a cost-effective service chain construction utilizing a VNF sharing model. Our proposed method employs an M/M/1/K queuing model to represent each VNF, allowing us to assess the relationship between resource allocation and loss probability. The method makes decisions regarding VNF sharing, VNF placement, amount of resources for each VNF, and the transmission route of each service chain. These decisions are guided by our proposed heuristic algorithm, designed to address the optimization problem. The performance of the proposed method is assessed through simulation. The key contribution of this study is summarized as follows.

- We model packet processing within a VNF using an M/M/1/K finite capacity queue to account for packet loss probability.
- We formulate an optimization problem aimed at determining VNF placement, the allocated resources for VNFs, and the transmission path of each service chain to minimize the cost of service chain construction.
- To solve the optimization problem, we propose a heuristic service chain construction algorithm.

3.1 System Model

In this section, we delineate a system model and introduce variables pertinent to the system model.

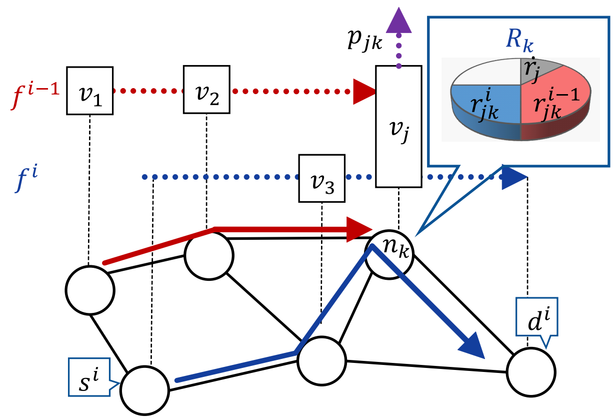

Figure 3.1: System model.

Table 3.1: Symbols for system model.

$\mathcal{N}$	Set of nodes
n_k	The kth node
$\mathcal{E}$	Set of links
e_{kl}	Link between n_k and n_l
R_k	Amount of available resources in n_k
B_{kl}	Maximum bandwidth of e_{kl}
$\mathcal{V}$	Set of VNFs
v_j	The jth VNF
$\mathcal{F}$	Set of user requests for service chain construction
f^i	Service chain of the ith user's request
s^i	Source node of f^i
d^i	Destination node of f^i
t^i_S	Amount of traffic of f^i injected from s^i
$\mathcal{V}^i$	Set of VNFs that are used in f^i
P^i	Requirement for packet loss probability for f^i
v^i_j	The jth VNF in $\mathcal{V}^i$
$\dot{r}_j$	Placement resources for v_j
r^i_{jk}	Processing resources for v_j to process packets of f^i in n_k
c^{res}_{jk}	Resource consumption costs for all v^i_j in $\mathcal{F}$ on n_k
p_{jk}	Loss probability of v_j in n_k

The system model is illustrated in Figure 3.1, and the variables used are summarized in Table 3.1. In this system model, a communication network is represented as a graph comprising a node set and a link set, denoted by $\mathcal{N} = \{n_k \,|\, k = 1, \cdots, |\mathcal{N}|\}$ and $\mathcal{E} = \{e_{kl} \,|\, n_k \in \mathcal{N}, n_l \in \mathcal{N}, k \neq l\}$, respectively. Here, a COTS server for a VNF is deployed on the kth node $n_k \in \mathcal{N}$, and the link between n_k and n_l is denoted as $e_{kl} \in \mathcal{E}$. Additionally, the amount of resources R_k is available for VNFs in n_k, and the bandwidth of e_{kl} is represented as B_{kl}.

Various types of VNFs can be used in this communication network and a set of VNFs is denoted as $\mathcal{V} = \{v_1, \cdots, v_{|\mathcal{V}|}\}$. A set of service chain requests is represented as $\mathcal{F}$, and the ith service chain request is denoted as $f^i \in \mathcal{F}$. The service chain f^i is denoted as $(s^i, d^i, t^i_S, \mathcal{V}^i, P^i)$, where s^i (d^i) represents the source (destination) node. Moreover, t^i_S denotes the amount of traffic injected from s^i, and $\mathcal{V}^i$ is a set of VNFs used in f^i. Here, when f^i utilizes VNF v_j, v_j is especially denoted as $v^i_j \in \mathcal{V}^i$. For f^i, a transmission route is chosen among K shortest paths, which are derived by the K shortest path algorithm, between s^i and d^i, and P^i is the requirement for packet loss probability. The model does not take into account link weights; therefore, the selection of K shortest paths is based on the number of hops.

In terms of the placement of VNFs, the amount of placement resources $\dot{r}_j$ is needed to place the jth VNF v_j in any node. Moreover, the amount of processing resources denoted as r^i_{jk} is needed to process packets of f^i with v_j, which is also represented as v^i_j, in n_k. Here, r^i_{jk} depends on the amount of traffic t^i_S. If v_j in node n_k can be shared by multiple service chains, the amount of placement resources is still $\dot{r}_j$ because the number of VNFs does not change. By contrast, the amount of processing resources is expected to be increased to process a larger number of packets. Here, the loss probability for v_j in n_k is denoted as p_{jk}, and p_{jk} must be equal to or smaller than P^i.

The cost for v_j in n_k is represented as c^{res}_{jk}, and is given by

$$c^{res}_{jk} = \min(1, \left\lceil \sum_{i=1}^{|\mathcal{F}|} r^i_{jk} \right\rceil)\dot{r}_j + \sum_{i=1}^{|\mathcal{F}|} r^i_{jk}. \tag{3.1}$$

In (3.1), the cost depends on the placement resources and the processing resources for each VNF. Here, $\min(g, h)$ is equal to g when g is smaller than h. By contrast, $\min(g, h)$ is equal to h when g is larger than h. If v_j is placed in n_k for one or more service chains, $\lceil \sum_{i=1}^{|\mathcal{F}|} r^i_{jk} \rceil$ is larger than or equal to one. As a result, $\min(1, \lceil \sum_{i=1}^{|\mathcal{F}|} r^i_{jk} \rceil)$ is equal to one, and hence the first term is $\dot{r}_j$. If v_j is not placed in n_k, $\min(1, \lceil \sum_{i=1}^{|\mathcal{F}|} r^i_{jk} \rceil)$ is equal to zero and the first term is also equal to zero. By sharing VNFs among multiple service chains, the value of the first term in (3.1) decreases, whereas the second term in (3.1) increases. Here, c^{res}_{jk} is the cost for v_j in terms of management, energy consumption, and so on, in n_k, and we assume that the cost is equal to the amount of resources for v_j in n_k.

Table 3.2: Symbols for optimization problem.

x^i_j	Amount of processing v_j for f^i
y^i_{jk}	Index variable for placing v_j in n_k for f^i
z^{iq}	Index variable for selecting the qth shortest path between s^i and d^i for f^i
δ^{iq}_k	Index variable for relationship n_k and the qth shortest path for f^i
ϵ^{iq}_{kl}	Index variable for the relationship between link e_{kl} and the qth shortest path for f^i
t^i_{kl}	Amount of traffic transmitted over e_{kl} for f^i without packet loss
λ_{jk}	Arrival rate for v_j in n_k
μ_{jk}	Service rate for v_j in n_k
K_{jk}	System capacity for v_j in n_k
$p_{K_{jk}}$	Probability of arriving a packet in the system that has K customers for v_j in n_k

3.2 Optimization Problem Formulation for Service Chain Construction

In this section, we address the system model presented in Sect. 3.1 by presenting a formulation of an optimization problem. The objective is to construct service chains with the aim of minimizing costs while meeting the QoS requirement concerning the loss probability. The variables relevant to the optimization problem are detailed in Table 3.2.

The following focuses on the ith service chain f^i and describes three variables. The first variable, x^i_j, is the amount of processing resources of v_j for f^i, and the second one is expressed as follows:

$$y^i_{jk} = \begin{cases} 1, & \text{if VNF } v^i_j \text{ is placed in node } n_k, \\ 0, & \text{otherwise.} \end{cases} \tag{3.2}$$

The third variable is expressed as follows:

$$z^{iq} = \begin{cases} 1, & \text{if the route for } f^i \text{ is the } q\text{th shortest path} \\ & \text{between } s^i \text{ and } d^i, \\ 0, & \text{otherwise.} \end{cases} \tag{3.3}$$

Regarding the network environment, we define some variables for f^i as follows. The variable δ_k^{iq} denotes whether the node n_k is included in the qth shortest path, and is expressed as follows:

$$\delta_k^{iq} = \begin{cases} 1, & \text{if the node } n_k \text{ is included in} \\ & \text{the } q\text{th shortest path for } f^i, \\ 0, & \text{otherwise.} \end{cases} \tag{3.4}$$

The variable ϵ_{kl}^{iq} represents whether the link e_{kl} is included in the qth shortest path, and is denoted as follows:

$$\epsilon_{kl}^{iq} = \begin{cases} 1, & \text{if the link } e_{kl} \text{ is included in} \\ & \text{the } q\text{th shortest path for } f^i, \\ 0, & \text{otherwise.} \end{cases} \tag{3.5}$$

Then, the amount t_{kl}^i of traffic that is transmitted over e_{kl} is represented as follows:

$$t_{kl}^i = \sum_{q=1}^{K} t_S^i \, z^{iq} \, \epsilon_{kl}^{iq}. \tag{3.6}$$

Moreover, r_{jk}^i denotes the amount of processing resources for v_j^i in n_k, and is expressed as follows:

$$r_{jk}^i = x_j^i y_{jk}^i. \tag{3.7}$$

If v_j is not placed for f^i in n_k, that is, y_{jk}^i is zero, r_{jk}^i is also zero. From the above, the resource consumption cost c_{jk}^{res} for the placement of v_j in n_k is calculated with (3.1).

From these variables, we formulate an optimization problem for service chain construction to minimize the cost as follows.

$$\min_{x,y,z} \quad \sum_{j=1}^{|\mathcal{V}|} \sum_{k=1}^{|\mathcal{N}|} c_{jk}^{res}, \tag{3.8}$$

subject to:

$$x_j^i \geq 0, \quad \forall f^i \in \mathcal{F}, \forall v_j \in \mathcal{V}, \tag{3.9}$$

$$\sum_{k=1}^{|\mathcal{N}|} y_{jk}^i = 1, \quad \forall f^i \in \mathcal{F}, \forall v_j \in \mathcal{V}, \tag{3.10}$$

$$\sum_{q=1}^{K} z^{iq} = 1, \quad \forall f^i \in \mathcal{F}, \tag{3.11}$$

$$\sum_{j=1}^{|\mathcal{V}|} \sum_{k=1}^{|\mathcal{N}|} \sum_{q=1}^{K} y_{jk}^i z^{iq} \delta_k^{iq} = |\mathcal{V}^i|, \quad \forall f^i \in \mathcal{F}, \tag{3.12}$$

$$\sum_{i=1}^{|\mathcal{F}|} t_{kl}^i \leq B_{kl}, \quad \forall e_{kl} \in \mathcal{E}, \tag{3.13}$$

$$\sum_{j=1}^{|\mathcal{V}|} c_{jk}^{res} \leq R_k, \quad \forall n_k \in \mathcal{N}, \tag{3.14}$$

$$y_{jk}^i p_{jk} \leq P^i, \quad \forall f^i \in \mathcal{F}, \forall v_j^i \in \mathcal{V}^i, \forall n_k \in \mathcal{N}, \tag{3.15}$$

$$h_m^i < h_{m+1}^i, \quad 1 \leq m \leq |\mathcal{V}_i| - 1, \forall f^i \in \mathcal{F}. \tag{3.16}$$

The objective function (3.8) minimizes the total cost for all service chains. The constraint (3.9) states that the amount of processing resources for v_j used by f^i is greater than or equal to zero. The constraint (3.10) guarantees that v_j for the ith service chain f^i, which is denoted as v_j^i, must be placed on only one node. Additionally, the constraint (3.11) guarantees that only one route is selected among the K shortest paths as a transmission route for f^i. The constraint (3.12) indicates that v_j^i must be placed on the node in the qth route, and the constraint (3.13) indicates that the amount of traffic on e_{kl} is equal to or smaller than the link's bandwidth. Additionally, (3.14) ensures that VNFs cannot be placed in n_k using more than the maximum amount of resources R_k. According to the constraint (3.15), the requirement of f^i in terms of the loss probability is set for each VNF on n_k. Finally, (3.16) ensures the processing sequence of VNFs. Here, h^i_m denotes the number of hops between s^i and a node where the mth VNF in $\mathcal{V}^i$ is placed.

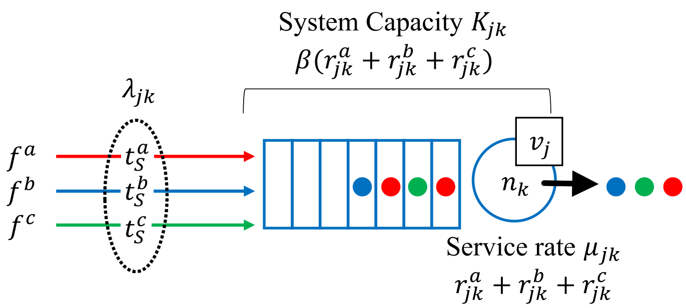

Figure 3.2: M/M/1/K queueing model for the proposed method.

As expressed in (3.1), the number of VNF instances is one, even if multiple service chains make use of VNF v_j in node n_k. Consequently, the loss probability p_{jk} is computed using an M/M/1/K queuing model. The queuing model is useful for calculating the packet loss probability when VNFs handle packets from multiple service chains. In an M/M/1/K queuing model, the system has k customers with the probability

$$p_k = \begin{cases} \frac{(1-\rho)\rho^k}{1-\rho^{K+1}}, & \rho \neq 1, \\ \frac{1}{K+1}, & \rho = 1, \end{cases} \tag{3.17}$$

where ρ, which is the utilization factor, is given by $\rho = \frac{\lambda}{\mu}$ from the arrival rate λ and the service rate μ. From the Poisson arrivals see time averages, the packet loss probability in M/M/1/K is represented as p_K.

In our system model, as shown in Fig. 3.2, the arrival rate λ_{jk}, the service rate μ_{jk}, and the system capacity K_{jk} for the jth VNF in a node n_k are given by

$$p_{jk} = \begin{cases} \frac{(1-\rho_{jk})(\rho_{jk})^{K_{jk}}}{1-\rho_{jk}^{K_{jk}+1}}, & \rho_{jk} \neq 1, \\ \frac{1}{K_{jk}+1}, & \rho_{jk} = 1, \end{cases} \tag{3.18}$$

$$\rho_{jk} = \frac{\lambda_{jk}}{\mu_{jk}}, \tag{3.19}$$

$$\lambda_{jk} = \sum_{i=1}^{|\mathcal{F}|} t_S^i y_{jk}^i, \tag{3.20}$$

$$\mu_{jk} = \sum_{i=1}^{|\mathcal{F}|} r_{jk}^i, \tag{3.21}$$

$$K_{jk} = \gamma \sum_{i=1}^{|\mathcal{F}|} r_{jk}^i, \tag{3.22}$$

where γ is a parameter that denotes the relationship between the total amount of processing resources and the buffer for each VNF. Additionally, $r_{jk}^i = x_j^i y_{jk}^i$ in (3.22).

3.3 Heuristic Service Chain Construction Algorithm for Total Cost Minimization

In this section, we propose a heuristic service chain construction algorithm for minimizing the cost of the service chain construction. Please see variables in Table 3.3 to understand the heuristic algorithm, as well as those in Tables 3.1 and 3.2.

3.3.1 Overview

Table 3.3: Symbols for heuristic algorithm.

x_j^{iq}	Amount of resources v_j of f^i in the qth shortest path
y_{jk}^{iq}	Index variable for placing v_j for f^i in n_k in the qth shortest path
$\mathcal{N}^{iq}$	Set of nodes in the qth shortest path for f^i
$\mathcal{E}^{iq}$	Set of links in the qth shortest path for f^i
r_k	Amount of remaining resources in n_k
t_{kl}	Total amount of traffic on e_{kl}
c^{iq}	Amount of estimated cost for f^i in the qth shortest path
I_j^{iq}	Index variable for placing v_j for f^i in the qth shortest path

Algorithm 1: Service Chain Construction Algorithm for Optimization Problem

1 **Input :** $\mathcal{F}$
2 **Output :** $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$
3 **for** $i = 1$ *to* $|\mathcal{F}|$ **do**
4   **for** $q = 1$ *to* K **do**
5   **if** *RouteCheck*(f^i, q) *is true* **then**
              /*Algorithm 2*/
6   **for** *all* $v_j \in \mathcal{V}^i$ **do**
7   x_j^{iq} is calculated based on $t_S^i y_{jk}^i$, r_{jk}^i, and γr_{jk}^i in Eqs. (3.17) - (3.22)
8   VNFPlacement(f^i, q, x_j^{iq}) /*Algorithm 3*/
9   RouteDetermination$(f^i, x_j^{iq}, y_{jk}^{iq})$ /*Algorithm 4*/
10 x_j^i is allocated based on λ_{jk}, μ_{jk}, and K_{jk} in Eqs. (3.17) - (3.22)
11 Service chains are constructed based on $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$, and $\mathcal{F}$

In the following, from the three decision variables x_j^i, y_{jk}^i, and z^{iq} that were defined in section 3.2, the three sets $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$ are defined as follows:

$$\mathcal{X} = \{x_j^i \mid 1 \leq i \leq |\mathcal{F}|,\ 1 \leq j \leq |\mathcal{V}|\}, \tag{3.23}$$

$$\mathcal{Y} = \{y_{jk}^i \mid 1 \leq i \leq |\mathcal{F}|,\ 1 \leq j \leq |\mathcal{V}|, 1 \leq k \leq |\mathcal{N}|\}, \tag{3.24}$$

$$\mathcal{Z} = \{z^{iq} \mid 1 \leq i \leq |\mathcal{F}|,\ 1 \leq q \leq K\}. \tag{3.25}$$

Our proposed heuristic algorithm derives these decision variables for f^i.

The proposed algorithm consists of four algorithms. In the main algorithm, which is Algorithm 1, the bandwidth of the K shortest paths is checked for f^i with the function *RouteCheck* on line 5. RouteCheck is described in Algorithm 2 in the next section. Then, on line 7, the amount of resources x_j^{iq} is calculated.

Next, on line 8, the function *VNFPlacement* is processed for determining the VNF placement. VNFPlacement is explained in Algorithm 3 in section 3.3.3. On line 9, a route is determined among the K shortest paths with the function *RouteDetermination*, which is described in Algorithm 4 in section 3.3.4. Finally, x_j^i is allocated for each service chain, and the service chain construction is complete on line 11.

In the following, we describe each algorithm in more detail.

3.3.2 Route Check

Algorithm 2: RouteCheck

```
1 Input : f^i, q
2 Output : true or false
3 for all e_kl ∈ E^iq do
4     if t^i_S > B_kl − t_kl then
5         return false
6         /*The qth shortest path does not have sufficient bandwidth.*/
7 return true
```

Algorithm 2 (RouteCheck) checks whether the amount of traffic transmitted in the qth shortest path is equal to or larger than t^i_S. In the following, $\mathcal{E}^{iq}$ denotes a set of links in the qth shortest path for f^i.

In this algorithm, on line 4, it is checked whether the amount of traffic for f^i, t^i_S, can be accommodated on e_{kl} by comparing t^i_S with the amount of available bandwidth, $B_{kl} - t_{kl}$, where $t_{kl} = \sum_{\xi=1}^{i-1} t^{\xi}_{kl}$. Note that this algorithm considers the lower bound for the amount of available bandwidth because some service chains can use other path that does not include e_{kl} in the actual service chain construction. From lines 3-6, if the comparison of the bandwidth is true for all e_{kl}, f^i can be accommodated in the qth path. Otherwise, through the process on line 5, the qth shortest path can not be used for f^i.

3.3.3 Placement of VNFs

Algorithm 3: VNFPlacement

Input : f^i, q, x_j^{iq}

Output : y_{jk}^{iq}

/*Sharing VNF Placement*/

for *all* $v_j \in \mathcal{V}^i$ **do**

    for *all* $n_k \in \mathcal{N}^{iq}$ **do**

        if v_j *exists on* n_k *and* $y_{jk}^{iq} = 0$ **then**

            The amount r_k of remaining resource is calculated

            $y_{jk}^{iq} \leftarrow 1$

            if $OrderCheck(y_{jk}^{iq})$ **then**

                $y_{jk}^{iq} \leftarrow 0$

                continue

                /*The processing order is not satisfied.*/

            else if $x_j^{iq} > r_k$ **then**

                $y_{jk}^{iq} \leftarrow 0$

                continue

                /*n_k does not have sufficient resources.*/

17 /*Non-Sharing VNF Placement*/;
18 **for** *all* $v_j \in \mathcal{V}^i$ **do**
19   **for** *all* $n_k \in \mathcal{N}^{iq}$ **do**
20   **if** $y^{iq}_{jk} = 0$ **then**
21             $y^{iq}_{jk} \leftarrow 1$;
22   **if** $OrderCheck(y^{iq}_{jk})$ **then**
23                 $y^{iq}_{jk} \leftarrow 0$;
24   **continue;**
25             $y^{iq}_{jk} \leftarrow 0$;
26   The amount of remaining resource r_k is calculated ;
27   **if** $maxr < r_k$ **then**
28                 $maxr \leftarrow r_k$;
29                 $maxk \leftarrow k$;
30   /*Store the node that has more resources.*/;
31   **if** $x^{iq}_j < maxr$ **then**
32         $y^{iq}_{j\,maxk} \leftarrow 1$;
33         $I^{iq}_j \leftarrow 1$;
34   /*Place in the node that has more resources.*/;
35 /*If all VNFs in $\mathcal{V}^i$ are not placed, the VNFs are relocated.*/;
36 **if** $PlacementCheck(f^i)$ **then**
37   **for** *all* $v_j \in \mathcal{V}^i$ **do**
38   **for** *all* $n_k \in \mathcal{N}^{iq}$ **do**
39             $y^{iq}_{jk} \leftarrow 0$;
40   Execute the same process on lines 17-34

Algorithm 3 (VNFPlacement) determines the placement of VNFs for f^i in the qth shortest path that can accommodate f^i from Algorithm 2.

As mentioned in the previous section, $|\mathcal{F}|$ service chain requests are constructed while considering the VNFs shared among the multiple service chains to reduce the amount of VNF resources. In the optimization problem (3.8)-(3.10), all VNFs in $\mathcal{V}^i$ are placed optimally by considering all service chains at the same time. However, not all VNFs in $\mathcal{V}^i$ can be considered simultaneously in the heuristic algorithm, and service chain f^i has to be constructed without consideration of service chains f^{i+1} through $f^{|\mathcal{F}|}$. Therefore, from lines 4-16, only the VNFs that have already been placed for f^1 to f^{i-1} can be considered for the construction of f^i. In this VNF placement, if the same type of VNF has already been placed, f^i shares the VNF with other service chains. On lines 9-13, the order of VNFs and the amount of resources for each VNF are checked, respectively. Figure 3.3 shows an example of processing on lines 3-16 in Algorithm 3. In this figure, v_j for f^{i-1} is shared by f^i; however, v^3 is not shared because v^3 has not been placed.

Then, other VNFs in $\mathcal{V}^i$ are placed on lines 17-34. On lines 26-30, *maxk* is the index of the node whose amount of available resources is the maximum, and *maxr* is its amount of resources. VNFs are placed in nodes having more resources. On lines 31-34, the VNFs are placed in n_{maxk}. On lines 35-40, all VNFs in $\mathcal{V}^i$ are replaced if not all the VNFs can be placed.

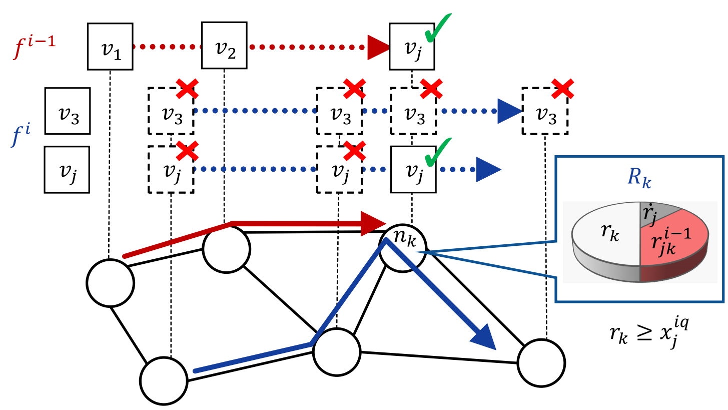

Figure 3.3: Example of processes on lines 3-16 in Algorithm 3.

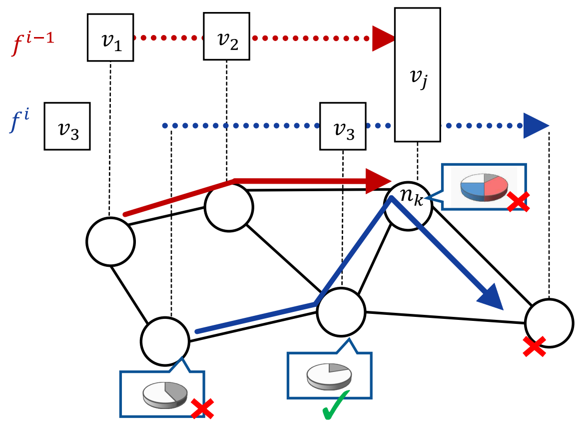

Figure 3.4: Example of processes on lines 17-34 in Algorithm 3.

Figure 3.4 shows an example of processes on lines 17-34 in Algorithm 3. In this figure, v_3 is placed for f^i because it cannot be placed in Fig. 3.3. Here, v_3 cannot be placed in n_k because the amount of resources is inadequate, and it is thus placed in another node.

3.3.4 Determination of Route

Algorithm 4: RouteDetermination

Input : $f^i, x_j^{iq}, y_{jk}^{iq}$
Output : $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$
for $q = 1$ *to* K **do**
    for *all* $n_k \in \mathcal{N}^{iq}$ **do**
        for *all* $v_j \in \mathcal{V}^i$ **do**
            $c_{jk}^{iq} \leftarrow I_j^{iq} \dot{r}_j + x_j^{iq} y_{jk}^{iq}$
            $c^{iq} \leftarrow c^{iq} + c_{jk}^{iq}$
Select index q^* that has the minimum value of c^{iq}
for *all* $v_j \in \mathcal{V}^i$ **do**
    $x_j^i \leftarrow x_j^{iq^*}$
    for *all* $n_k \in \mathcal{N}^{iq}$ **do**
        $y_{jk}^i \leftarrow y_{jk}^{iq^*}$
$z^{iq^*} \leftarrow 1$

Algorithm 4 determines the optimal route among the K shortest path based on (3.8). Initially, the cost of f^i is calculated from lines 3-7. This algorithm then selects the q^*th shortest path that has the minimum cost as the optimal route. From lines 9-12, for the q^*th shortest path, x^i_j, y^i_{jk}, and z^{iq} are updated. Here, on line 6, I^{iq}_j denotes that v_j is placed for f^i in the qth shortest path. Moreover, c^{iq} represents the cost for f^i in the qth shortest path.

3.4 Numerical Examples

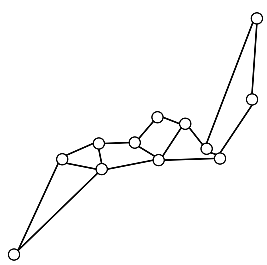

Figure 3.5: JPNM.

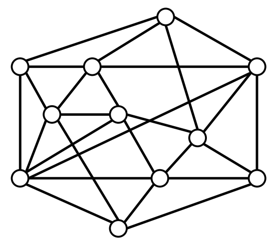

Figure 3.6: COST239.

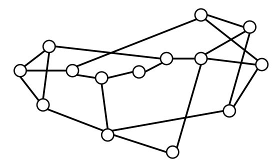

Figure 3.7: NSFnet.

In this section, we evaluate the performances of our proposed cost-effective service chain construction under three typical network topologies: Japan Photonic Network Model (JPNM) [46], COST239 [47], and NSFnet [48]. The JPNM topology consists of 12 nodes and 17 links, and the COST239 topology consists of 11 nodes and 26 links. Moreover, the NSFnet topology consists of 14 nodes and 21 links. These topologies are effective to investigate the performance of our proposed method in general situations and use cases.

In these networks, five types of VNFs ($\mathcal{V} = \{v_1, \cdots, v_5\}$) can be utilized for each service chain, and the maximum amount of available resources R_k in node n_k is 1000. The bandwidth B_{kl} of link e_{kl} is equal to 1000 for any pairs of nodes n_k and n_l. When $v_j \in \mathcal{V}$ is placed in a node, a fixed amount of resources $\dot{r}_j$ are needed, and $\dot{r}_j$ is equal to 10 regardless of the type of VNF. For the ith service chain f^i, a source node s^i and a destination node d^i are selected among all nodes at random. The number of VNFs $|\mathcal{V}_i|$ that are used in f^i is equal to three, and three types of VNFs are selected among the five types of VNFs in $\mathcal{V}$ at random. In addition, the amount t^i_S of traffic is set to 30, and the number K of shortest paths is set to 3.

In this scenario, we evaluate the performance of our heuristic service chain construction algorithm described in Section 3.3. Moreover, we evaluate a method that can solve the optimization problem shown in Section 3.2 by using the genetic algorithm (GA). For a performance comparison, we also evaluated the performances of two methods, i.e., the independent chain processing (ICP) and parameter allocating resources (PAR) methods.

With the ICP method, the amount of resources for the processing is calculated and allocated for each VNF to satisfy the requirements in terms of the loss probability in each service chain independently even if that VNF is shared by multiple service chains. Note that the consumption of the amount $\dot{r}_j$ of resources is suppressed by sharing the VNF. Therefore, (3.15) is replaced as follows:

$$p^i_{jk} \leq P^i, \ \forall f^i \in \mathcal{F}, 1 \leq j \leq |\mathcal{V}|, 1 \leq k \leq |\mathcal{N}|. \tag{3.26}$$

The ICP method calculates the loss probability p^i_{jk} based on (3.17) with $\rho = \frac{t^i_S y^i_{jk}}{r^i_{jk}}$ and $K = \gamma r^i_{jk}$.

The PAR method does not use the queueing model to calculate the loss probability for the resource allocation. The PAR method uses the parameter α to determine the amount of processing resources for each VNF. Therefore, (3.15) is replaced as follows:

$$r^i_{jk} \geq (1+\alpha)t^i_S, \ \ \forall f^i \in \mathcal{F}, 1 \leq j \leq |\mathcal{V}|, 1 \leq k \leq |\mathcal{N}|. \tag{3.27}$$

In the following, α is set to 0.01, 0.05, 0.1, and 0.2. With this method, more processing resources r^i_{jk} are allocated based on α allowing t^i_S to adjust the loss probability of the VNFs.

3.4.1 Impact of Number of Service Chains

In this section, we evaluate the impact of the number of service chains on the objective function in terms of the total cost for our proposed method, the GA method, the ICP method, and the PAR method. Moreover, the requirement in terms of the loss probability P^i is set to 0.001.

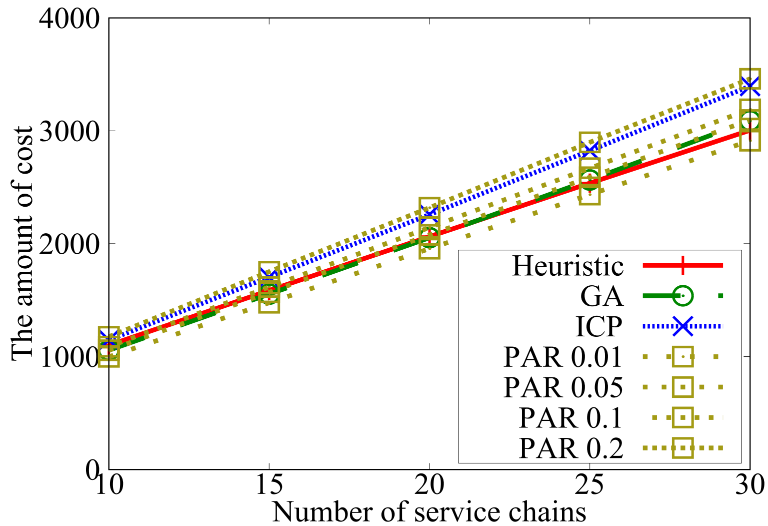

Figure 3.8: Value of objective function vs. number of service chains for JPNM.

Figure 3.8 shows how the number of service chains affects the total cost of the VNFs for the four methods. From this figure, we find that the total cost for all methods is increased with an increase in the number of service chains. This is because the amount of resources for each VNF increases to process more packets. This means that λ in (3.20) increases and a larger amount of resources are required to satisfy P^i.

We find that the total costs of the Heuristic and GA methods are consistently smaller than the IP and PAR methods. In addition, we can see that the result for the Heuristic method is much closer to that of the GA method. Although the cost of the PAR method may be smaller than that of our proposed approach, the PAR method constructs the service chains without considering the loss probability. Therefore, in some cases of α, the PAR method may not satisfy the requirement in terms of the loss probability (3.15).

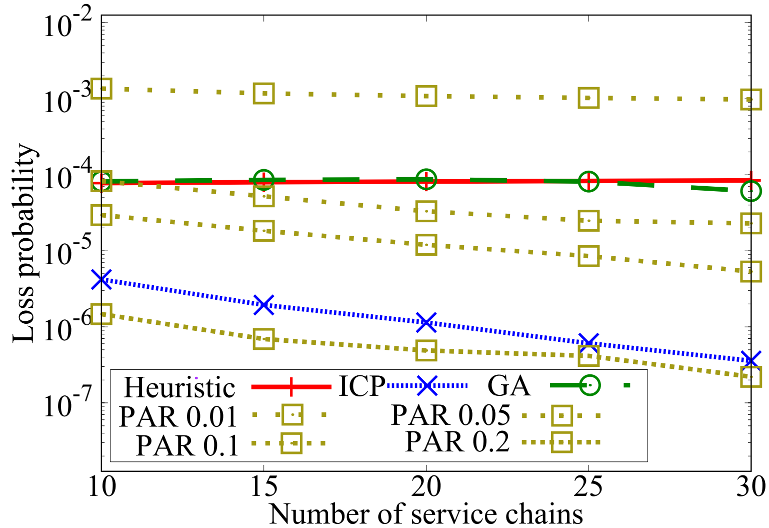

Figure 3.9: Loss probability p_{jk} vs. number of service chains for JPNM.

Figure 3.9 shows how the number of service chains affects the loss probability p_{jk} for the four methods. Note that the requirement in terms of the loss probability P^i is set to 0.001. From this figure, we find that our proposed method keeps the loss probability p_{jk} constant despite the number of service chains. By contrast, the ICP and PAR methods decrease p_{jk} with an increase in the number of service chains. This is because the ICP method allocates the processing resources to each VNF independently, even if the VNFs are shared among the multiple service chains. Moreover, because the PAR method allocates the processing resources for the VNFs according to the parameter α only, it is difficult to set α appropriately to satisfy P^i.

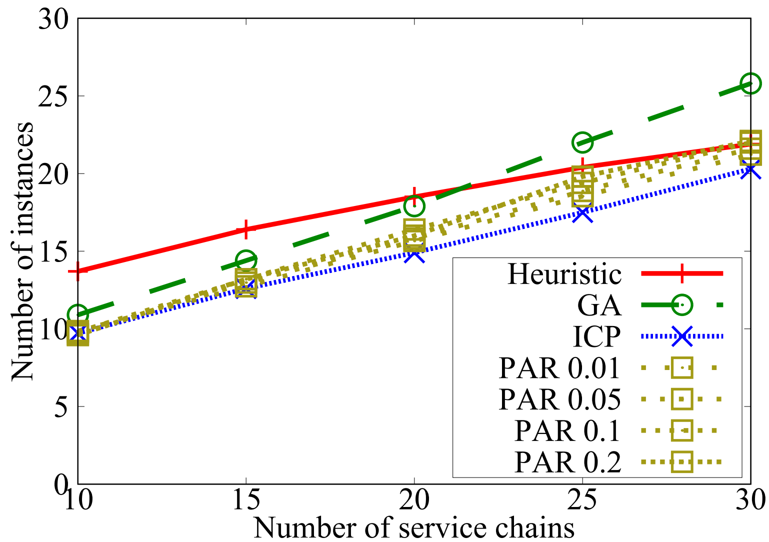

Figure 3.10: Number of VNF instances vs. number of service chains for JPNM.

Figure 3.10 shows how the number of service chains affects the number of VNF instances for the four methods. From this figure, we find that the number of VNF instances for all methods increases with an increase in the number of service chains. However, we can see that each method clearly suppresses the number of VNF instances by comparing with the sum of $\sum_{i=1}^{|\mathcal{F}|} |\mathcal{V}^i|$. As shown in (3.1), the number of VNF instances directly affects the cost of the VNFs. Therefore, all methods attempt to share the VNFs among multiple service chains to suppress the cost. By contrast, although the Heuristic method has more VNF instances in comparison with other methods, the cost is suppressed, similar to the other methods. This is because $\dot{r}_j$ is set to 10, which is small for the total cost.

From these results for JPNM, we find that our proposed method can minimize the cost of the VNFs by sharing the VNFs among multiple service chains. Moreover, in terms of the loss probability, our proposed method can effectively allocate the resources for the processing by introducing the M/M/1/K queueing model such that p_{jk} is almost equal to P^i.

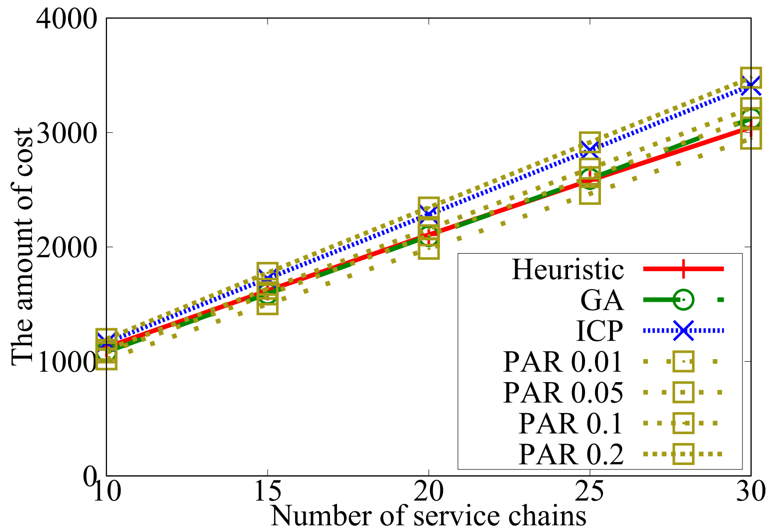

Figure 3.11: Value of objective function vs. number of service chains for COST239.

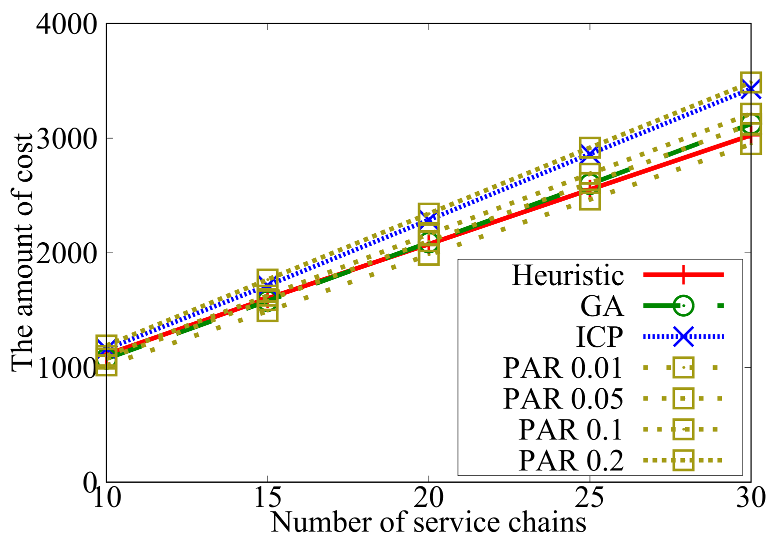

Figure 3.12: Value of objective function vs. number of service chains for NSFnet.

Figures 3.11 and 3.12 show the results for COST239 and NSFnet with same parameters used in JPNM. We observe the same trends as the results for JPNM in Fig. 3.8. From these results, we find that our proposed method can effectively construct the service chains regardless of the network topologies and the number of service chains. Moreover, with our proposed method, we can find that the performance of the Heuristic method is almost equal to that of the GA method.

3.4.2 Impact of Requirement in terms of Loss Probability

In this section, we evaluate the impact of the requirement in terms of the loss probability P^i on the objective function for the four methods. Moreover, the number of service chains is set to 20.

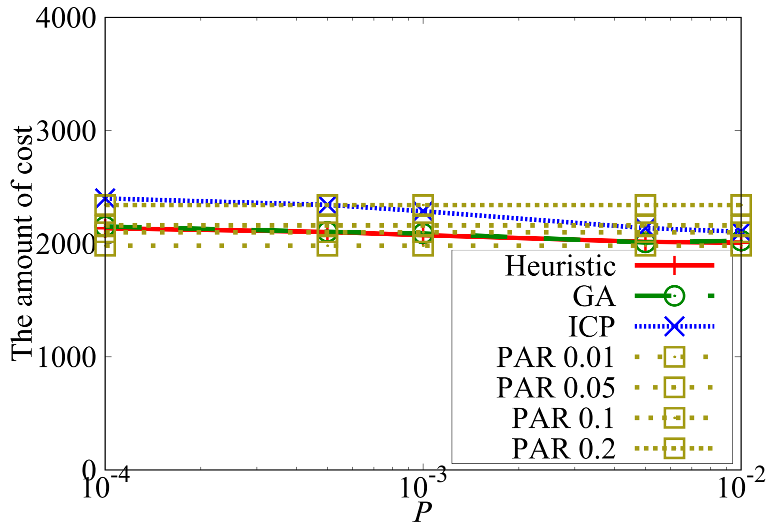

Figure 3.13: Value of objective function vs. requirement in terms of loss probability P^i for JPNM.

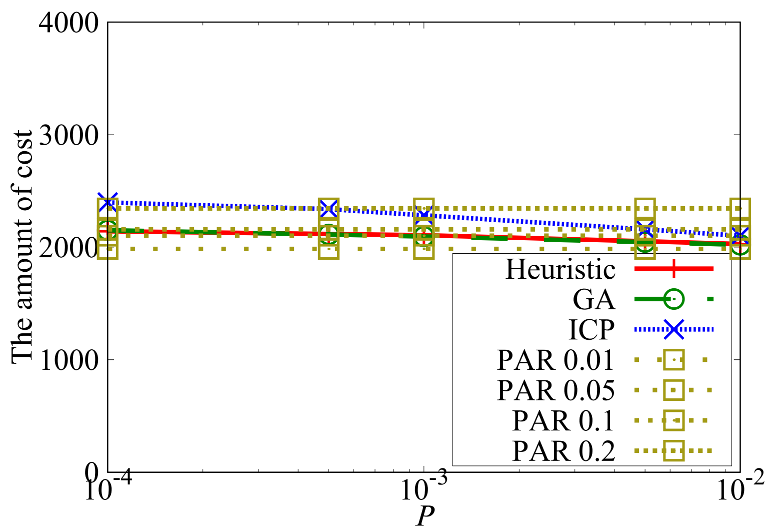

Figure 3.14: Value of objective function vs. requirement in terms of loss probability P^i for COST239.

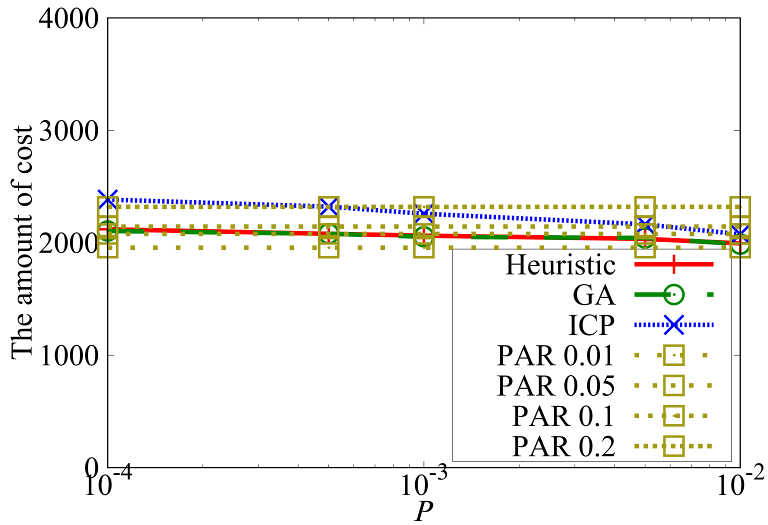

Figure 3.15: Value of objective function vs. requirement in terms of loss probability P^i for NSFnet.

Figures 3.13-3.15 show how the requirement in terms of the loss probability P^i affects the total cost of the service chain construction for the four methods. Note that the result of the PAR method is not changed regardless of P^i because it allocates the processing resources to each VNF according to the parameter α.

From these figures, we find that the cost for the ICP method and our proposed method increases with a decrease in the requirement in terms of the loss probability P^i. This is because the performance of each VNF is improved and the strict requirement in terms of the loss probability is satisfied by allocating more resources. Moreover, we find that the cost for the proposed method is smaller than that for the ICP method even if the requirement in terms of the loss probability becomes smaller. Thus, our proposed method can utilize the resources more effectively than the ICP method despite P^i.

As described in section 3.4.1, we find that the total costs of the Heuristic and GA methods are always smaller than those of the ICP method. In addition, we can see that the result for the Heuristic method is much closer to that of the GA method. From these results, we find that our proposed method can minimize the cost regardless of the requirements in terms of the loss probability P^i and the topologies.

3.4.3 Impact of Amount of traffic for a Service Chain

In this section, we evaluate the impact of the amount of traffic for a service chain on the objective function. Here, t_S^i is randomly selected between 30 and t_{max}. Moreover, the requirement in terms of the loss probability P^i is set to 0.001, and the number of service chains is set to 20.

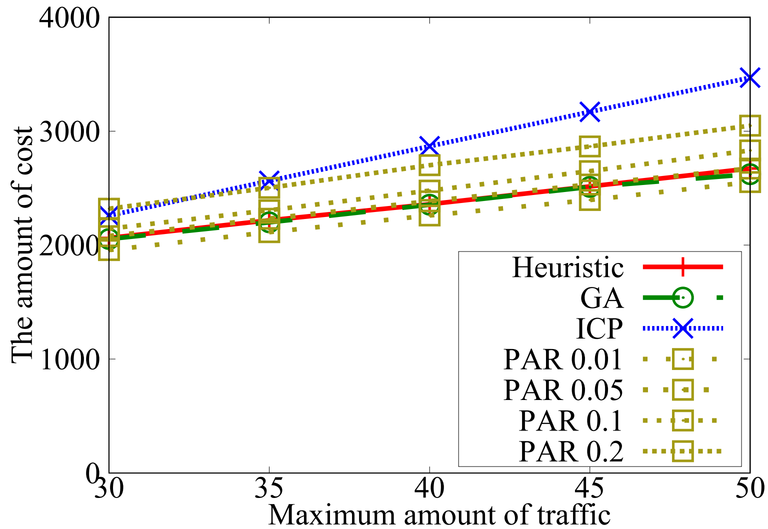

Figure 3.16: Value of objective function vs. maximum amount of traffic for a service chain for JPNM.

Figure 3.16 shows how the amount of traffic for a service chain affects the cost for the four methods. From this figure, we can see that the total cost for all methods increases with an increase in the amount of traffic for the service chains. This is because the amount of resources for each VNF increases to process more packets. However, we find that the total costs of the Heuristic and GA methods are always smaller than those of the ICP and PAR methods. In addition, we find that the result for the Heuristic method is much closer to that of the GA method. Note that the difference between our proposed method and the ICP approach is larger than the result in section 3.4.1 as indicated in the comparison between Figs. 3.16 and 3.8.

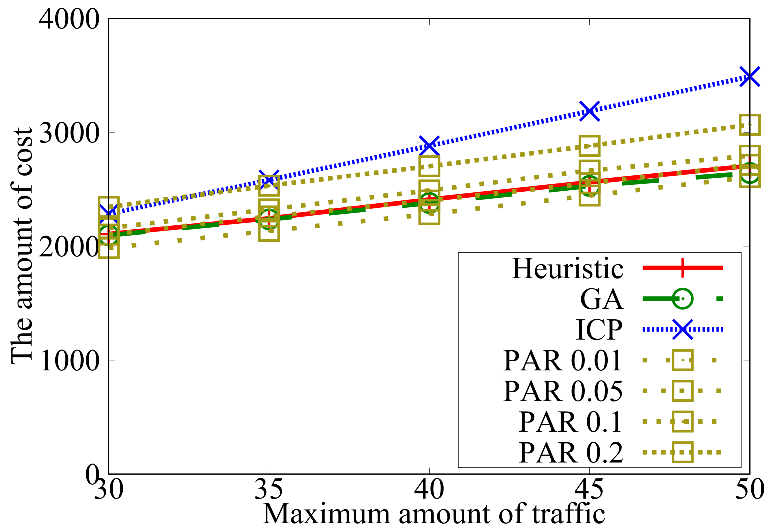

Figure 3.17: Value of objective function vs. maximum amount of traffic for a service chain for COST239.

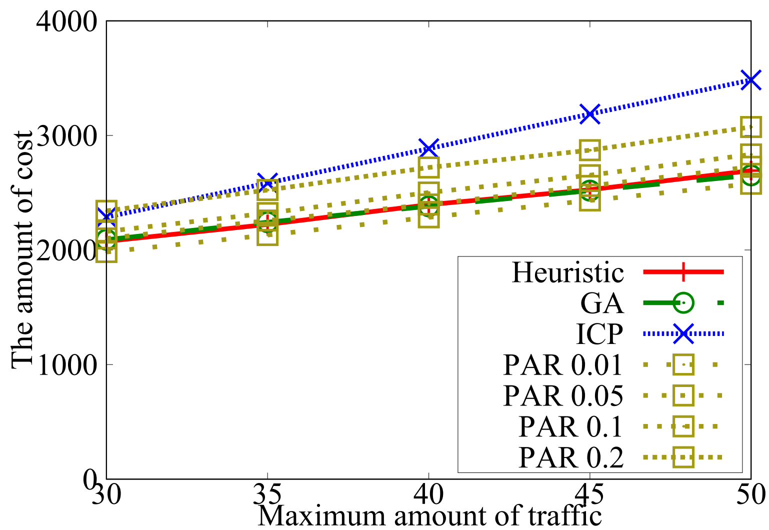

Figure 3.18: Value of objective function vs. maximum amount of traffic for a service chain for NSFnet.

Figures 3.17 and 3.18 show the results for COST239 and NSFNet with the same parameters as JPNM, respectively. We can see the same trends as in the results for JPNM in Fig. 3.16. From these results, we find that our proposed method can effectively construct the service chains regardless of the network topologies and the amount of traffic for a service chain.

3.4.4 Impact of Amount of Resources for a Node

In this section, we evaluate the impact of the minimum amount of resources for a node on the objective function for our proposed method, the ICP method, and the PAR method. The minimum amount of resources is represented as r_{min}, and R_k is randomly set to r_{min} or 1000. Moreover, P^i is set to 0.001, and the number of service chains is set to 20.

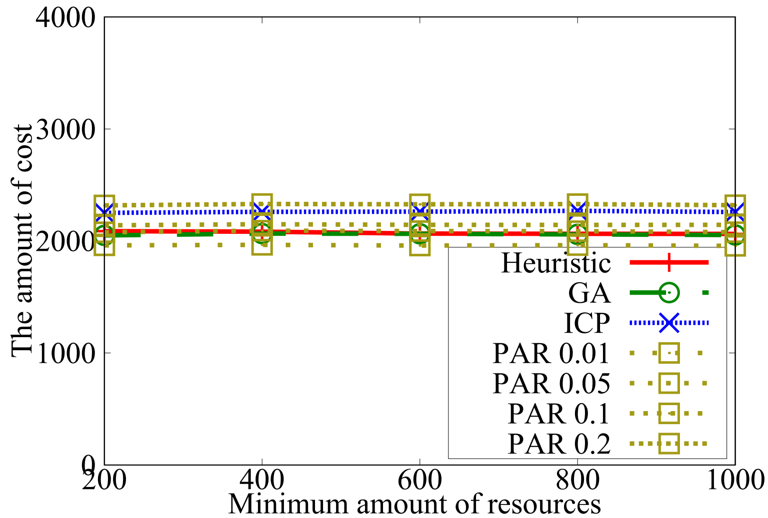

Figure 3.19: Value of objective function vs. minimum amount of resources for a node for JPNM.

Figure 3.19 shows how the amount of resources for a node affects the costs for the four methods. From this figure, we find that the costs for all methods are constant regardless of the amount of resources for a node. We also find that the costs of the Heuristic and GA methods are always smaller than those of the ICP and PAR methods. In addition, we find that the results for the Heuristic method are always much closer to those of the GA method.

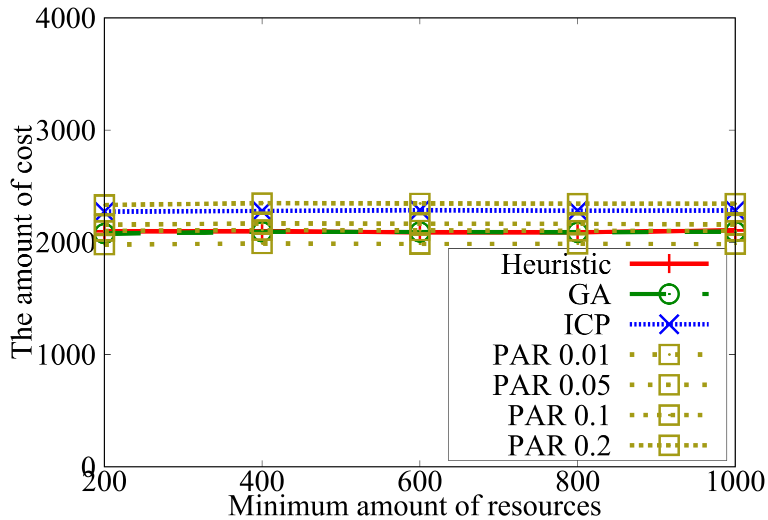

Figure 3.20: Value of objective function vs. minimum amount of resources for a node for COST239.

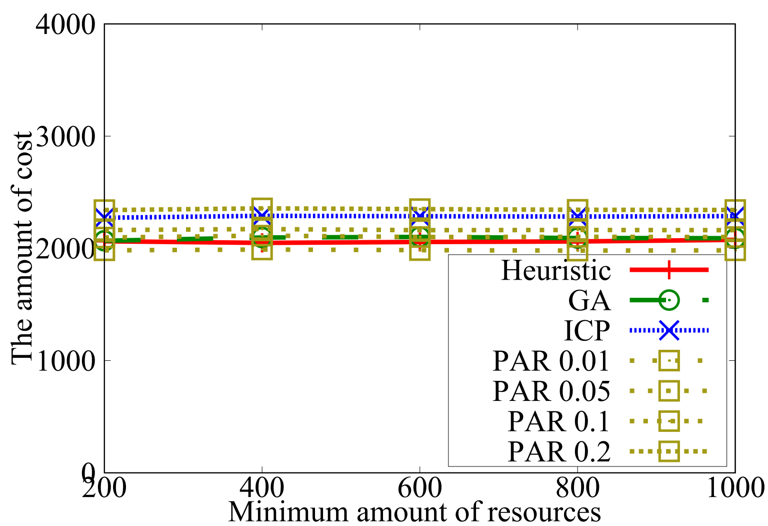

Figure 3.21: Value of objective function vs. minimum amount of resources for a node for NSFnet.

Figures 3.20 and 3.21 show the results for COST239 and NSFnet, respectively, with same parameters used in JPNM. We can see the same trends as the results for JPNM in Fig. 3.19. From these results, we find that our proposed method can effectively construct the service chains regardless of the network topologies and the amount of resources for a node.

From section 3.4.1 to 3.4.4, we can obtain similar results for different network topologies. This is because the transmission path is selected by K shortest path algorithm. The number of candidates for transmission path is limited to K, and hence the number of candidates for VNF placement nodes is decreased. As a result, the impact of network topologies on the proposed method is not so large. Moreover, the impact of sharing VNFs becomes small if the candidate of transmission path is not limited or K is large. Therefore, when the amount of traffic is small, the effectiveness of our proposed method is small. However, the effectiveness of our proposed method will increase as the amount of traffic becomes large.

Chapter 4

Heuristic-based Service Chain Construction with Security-Level Management

In this chapter, we propose a a heuristic-based service chain construction with security-level management. To construct many service chains, the proposed method increases the security level of VNFs and nodes with security-level management. It is important to note that the addition of security mechanisms incurs additional costs. Therefore, the cost optimization problem for constructing the service chains with the security-level management is formulated, and a heuristic algorithm for solving the optimization problem is proposed as our service chain construction method. We evaluate our heuristic-based proposed method in three network topologies and investigate its performance in various cases. The key contribution of this study is summarized as follows.

- This is the first study to focus on the security-level management for service chaining.
- This study introduces the security-level management concept to construct many service chains.
- The cost optimization problem for service chaining is formulated in this study, and a heuristic algorithm is proposed.

Table 4.1: Symbols for system model.

$l(v^i_j)$	The security level of v_j for f^i
$d(v^i_j)$	The security demand of v_j for f^i
$l(n_k)$	The security level of n_k
$d(n_k)$	The security demand of n_k

4.1 System Model

This section describes and introduces a system model and its variables.

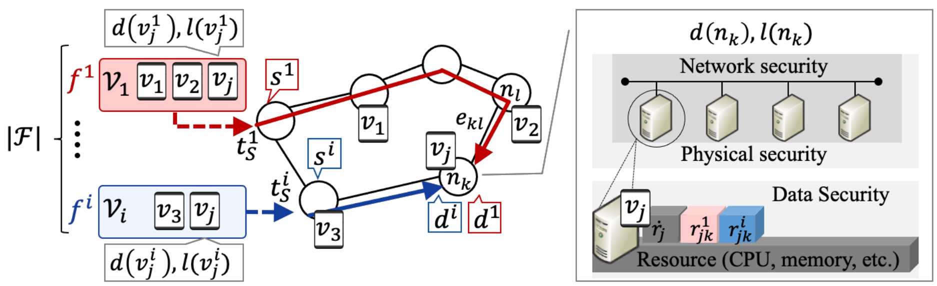

Figure 4.1: System model.

Figure 4.1 shows the system model, and Table 4.1 lists the variables used in the following. In addition to Chapter 3, the security is also considered for each VNF and each node with a server where VNFs are placed to construct service chains. Here, according to [22, 41], the security-level concept is introduced into each VNF and node. As shown in Fig. 4.1, we assume that the security level is represented as a value according to the effectiveness of the security mechanisms such as physical security, network security, and data security. The security levels of v^i_j and n_k are denoted as $l(v^i_j)$ and $l(n_k)$, respectively. Additionally, the security demands of v^i_j and n_k are denoted as $d(v^i_j)$ and $d(n_k)$, respectively. Many service chains must be accommodated in this network while satisfying the constraint regarding the amount of resources, the loss probability, and the security level.

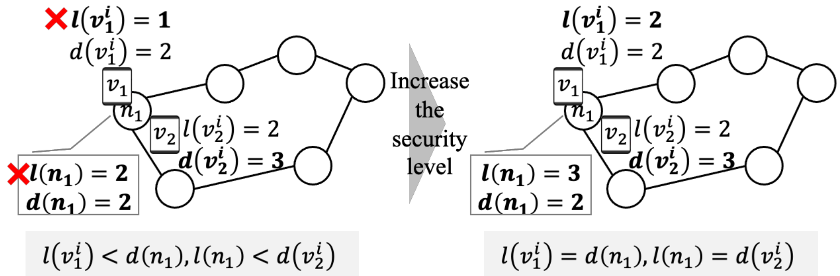

Figure 4.2: Security level and demand.

This model takes into account the security levels and demands of VNFs and nodes for constructing service chains as shown in Fig. 4.2. High-security VNF and nodes have more sophisticated security mechanisms, such as packet filters, access control, monitoring, and anti-virus. The security demand $d(v_j^i)$ of VNF v_j^i can be satisfied when v_j^i is installed on node n_k that can offer the same or a higher security-level $l(n_k)$. Similarly, the security demand $d(n_k)$ of n_k can be satisfied when v_j^i that can offer the same or a higher security-level $l(v_j^i)$ is installed on n_k. These relationships are expressed as follows:

$$l(v_j^i) \geq d(n_k), \quad \forall f^i \in \mathcal{F}, v_j^i \in \mathcal{V}^i, \forall n_k \in \mathcal{N}, \tag{4.1}$$

$$l(n_k) \geq d(v_j^i), \quad \forall f^i \in \mathcal{F}, v_j^i \in \mathcal{V}^i, \forall n_k \in \mathcal{N}. \tag{4.2}$$

This model assumes that each user determines the security demand $d(v_j^i)$ and level $l(v_j^i)$ of VNF v_j^i. For example, when a user sends the infrastructure provider a request of f^i, the user selects a preference for security from the pull-down menu, and the preference is automatically changed to $d(v_j^i)$ and $l(v_j^i)$.

When (4.1) and (4.2) are unsatisfied, new security mechanisms must be added to increase $l(v_j^i)$ and $l(n_k)$. This security-level increment increases the costs of the new security mechanisms. Therefore, to lower the cost of increasing the security level, (4.1) and (4.2) are updated as follows:

$$l(v_j^i) = d(n_k), \quad \forall f^i \in \mathcal{F}, v_j^i \in \mathcal{V}^i, \forall n_k \in \mathcal{N}, \tag{4.3}$$

$$l(n_k) = d(v_j^i), \quad \forall f^i \in \mathcal{F}, v_j^i \in \mathcal{V}^i, \forall n_k \in \mathcal{N}. \tag{4.4}$$

Here, when $l(v_j^i)$ of VNF v_j^i must be increased to $d(n_k)$ of node n_k, the cost $\alpha R_k\{d(n_k) - l(v_j^i)\}$ is needed to satisfy (4.3). However, the cost $\beta R_k\{d(v_j^i) - l(n_k)\}$

is needed to satisfy (4.4) in a case where $l(n_k)$ of n_k must be increased to $d(v^i_j)$ of v^i_j. Each of the two costs is presumptively proportional to the maximum amount R_k of the available resources of n_k. This is because a node with a large amount of resources requires more advanced security mechanisms. These costs remain the same as long as the difference between $l(v^i_j)$ and $d(n_k)$ ($l(n_k)$ and $d(v^i_j)$) is the same regardless of changes in security level and demand.

4.2 Optimization Problem Formulation for Service Chain Construction

Table 4.2: Symbols for the optimization problem.

η^i_j	Amount of the increased security level of v^i_j
θ_k	Amount of the increased security level of n_k
$c^{i,VNF}_{jk}$	Security costs for v^i_j on n_k
c^{node}_k	Security costs for n_k
δ^{iq}_k	Index variable for relationship n_k and the qth shortest path for f^i
ϵ^{iq}_{kl}	Index variable for the relationship between link e_{kl} and the qth shortest path for f^i

This section formulates an optimization problem for constructing service chains to minimize the total cost while satisfying the security level and demand for the system model described in Sect. 4.1. In addition to Chapter 3, variables for the optimization problem are shown in Tables 4.1 and 4.2.

The security level of some VNFs and nodes may be increased for the security-level management to satisfy (4.1) and (4.2). We consider two integer variables η^i_j and θ_k, which are expressed as follows:

$$\eta^i_j \geq 0,\ \forall f^i \in \mathcal{F},\ \forall v_j \in \mathcal{V}, \quad (4.5)$$

$$\theta_k \geq 0,\ \forall n_k \in \mathcal{N} \quad (4.6)$$

The VNF security cost for v^i_j is calculated using η^i_j, and is represented as follows:

$$c^{i,VNF}_j = \alpha \sum_{k=1}^{|\mathcal{N}|} R_k y^i_{jk} \eta^i_j. \quad (4.7)$$

Moreover, the node security cost for n_k is calculated using θ_k, and is expressed as follows:

$$c_k^{node} = \beta R_k \theta_k. \tag{4.8}$$

From these variables, an optimization problem for constructing service chains is formulated to minimize the total cost.

$$\min_{x,y,z,\eta,\theta} \sum_{j=1}^{|\mathcal{V}|} \sum_{k=1}^{|\mathcal{N}|} c_{jk}^{res} + \sum_{i=1}^{|\mathcal{F}|} \sum_{j=1}^{|\mathcal{V}^i|} c_j^{i,VNF} + \sum_{k=1}^{|\mathcal{N}|} c_k^{node}, \tag{4.9}$$

subject to:

$$x_j^i \geq 0, \ \ \forall f^i \in \mathcal{F}, \forall v_j \in \mathcal{V}, \tag{4.10}$$

$$\sum_{k=1}^{|\mathcal{N}|} y_{jk}^i = 1, \ \ \forall f^i \in \mathcal{F}, \forall v_j \in \mathcal{V}, \tag{4.11}$$

$$\sum_{q=1}^{K} z^{iq} = 1, \ \ \forall f^i \in \mathcal{F}, \tag{4.12}$$

$$\sum_{j=1}^{|\mathcal{V}|} \sum_{k=1}^{|\mathcal{N}|} \sum_{q=1}^{K} y_{jk}^i z^{iq} \delta_k^{iq} = |\mathcal{V}^i|, \ \ \forall f^i \in \mathcal{F}, \tag{4.13}$$

$$\sum_{i=1}^{|\mathcal{F}|} t_{kl}^i \leq B_{kl}, \ \ \forall e_{kl} \in \mathcal{E}, \tag{4.14}$$

$$\sum_{j=1}^{|\mathcal{V}|} c_{jk}^{res} \leq R_k, \ \ \forall n_k \in \mathcal{N}, \tag{4.15}$$

$$y_{jk}^i p_{jk} \leq P^i, \ \ \forall f^i \in \mathcal{F}, \forall v_j^i \in \mathcal{V}^i, \forall n_k \in \mathcal{N}, \tag{4.16}$$

$$l(v_j^i) + \eta_j^i \geq d(n_k), \ \ \forall f^i \in \mathcal{F}, \forall v_j^i \in \mathcal{V}^i, \forall n_k \in \mathcal{N}, \tag{4.17}$$

$$l(d_k) + \theta_k \geq d(v_j^i), \ \ \forall f^i \in \mathcal{F}, \forall v_j^i \in \mathcal{V}^i, \forall n_k \in \mathcal{N}, \tag{4.18}$$

$$h^i_m < h^i_{m+1}, \ \ 1 \le m \le |\mathcal{V}_i| - 1, \ \forall f^i \in \mathcal{F}. \tag{4.19}$$

The objective function (4.9) minimizes the total cost to construct all service chains. The constraint (4.10) ensure that the amount of processing resources for v_j used by f^i is greater than or equal to zero. The constraint (4.11) guarantees that v_j for the ith service chain f^i, which is denoted as v^i_j, must be placed on only one node. Moreover, the constraint (4.12) ensures that only one route is selected among the K shortest paths as a transmission route for f^i. The constraint (4.13) indicates that v^i_j must be placed on the node in the qth route, and the constraint (4.14) indicates that the amount of traffic on e_{kl} is equal to or smaller than the link's bandwidth. Additionally, (4.15) ensures that VNFs cannot be placed in n_k using more than the maximum amount of resources R_k. According to the constraint (4.16), the requirement of f^i in terms of the loss probability is set for each VNF on n_k. The constraints (4.17) and (4.18) ensure that (4.1) and (4.2) are satisfied using the security-level management, respectively. Finally, (4.19) ensures the processing sequence of VNFs. Here, h^i_m denotes the number of hops between s^i and a node where the mth VNF in $\mathcal{V}^i$ is placed.

Note that η^i_j and θ_k can be determined automatically from (4.3) and (4.4), respectively, to minimize the total cost. Therefore, η^i_j and θ_k can be excluded from the decision variables, and (4.17) and (4.18) can be omitted. However, this study formulates the optimization problem in this manner to allow for an expanded objective.

4.3 Heuristic Service Chain Construction Algorithm for Total Cost Minimization

This section proposes a heuristic service chain construction algorithm with optimal security-level management. Tables 4.1, 4.2, and 4.3 show the variables for our heuristic algorithm.

4.3.1 Overview

Table 4.3: Symbols for the heuristic algorithm.

x_j^{iq}	Amount of resources v_j of f^i in the qth shortest path
y_{jk}^{iq}	Index variable for placing v_j for f^i in n_k in the qth shortest path
η_j^{iq}	Amount of the increased security level of v_j^i in the q th shortest path
θ_k^q	Amount of the increased security level of n_k in the qth shortest path
$\bar{l}(v_j^i)$	Average of security level of VNFs for f^i
ζ	Unit resource volume
c_{total}^{iq}	Amount of the total cost for f^i in the qth shortest path
n_k	Candidate node for VNF placement

In the following, the five sets: $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$, $\mathcal{H}$, and Θ are defined based on the five decision variables x_j^i, y_{jk}^i, z^{iq}, η_j^i, and θ_k defined in Sect. 4.2.

$$\mathcal{X} = \{x_j^i \mid 1 \le i \le |\mathcal{F}|,\ 1 \le j \le |\mathcal{V}|\}, \tag{4.20}$$

$$\mathcal{Y} = \{y_{jk}^i \mid 1 \le i \le |\mathcal{F}|,\ 1 \le j \le |\mathcal{V}|, 1 \le k \le |\mathcal{N}|\}, \tag{4.21}$$

$$\mathcal{Z} = \{z^{iq} \mid 1 \le i \le |\mathcal{F}|,\ 1 \le q \le K\}. \tag{4.22}$$

$$\mathcal{H} = \{\eta_j^i \mid 1 \le i \le |\mathcal{F}|,\ 1 \le j \le |\mathcal{V}|\}. \tag{4.23}$$

$$\Theta = \{\theta_k \mid 1 \le k \le |\mathcal{N}|\}. \tag{4.24}$$

Here, $\mathcal{X}$ is a two-dimensional array, and the element x_j^i in the ith row and jth column is the amount of processing resources for v_j^i. $\mathcal{Y}$ is a three-dimensional array, and the element y_{jk}^i is one if v_j^i is placed in n_k. $\mathcal{Z}$ is a two-dimensional array, and the element z^{iq} in the ith row and qth column is one if f^i uses the qth shortest path. $\mathcal{H}$ is a two-dimensional array, and the element η_j^i in the ith row and jth column denotes the amount of increased security level of v_j^i. Θ is a one-dimensional array, and the kth element θ_k denotes the amount of increased security level of n_k. Our proposed heuristic algorithm derives these decision variables $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$, $\mathcal{H}$, and Θ for the optimization problem in Sect. 4 by updating x_j^i, y_{jk}^i, z^{iq}, η_j^i, and θ_k. Moreover, the proposed algorithm consists of four algorithms, and these algorithms are executed for K shortest paths of each service chain. Therefore, the four sets $\mathcal{X}^q$, $\mathcal{Y}^q$, $\mathcal{H}^q$, and Θ^q are defined in order to store the candidate for the decision variables in the qth path.

$$\mathcal{X}^q = \{x_j^{iq} \,|\, 1 \leq i \leq |\mathcal{F}|,\ 1 \leq j \leq |\mathcal{V}|,\ 1 \leq q \leq K\}, \tag{4.25}$$

$$\mathcal{Y}^q = \{y_{jk}^{iq} \,|\, 1 \leq i \leq |\mathcal{F}|,\ 1 \leq j \leq |\mathcal{V}|, 1 \leq k \leq |\mathcal{N}|,\ 1 \leq q \leq K\}, \tag{4.26}$$

$$\mathcal{H}^q = \{\eta_j^{iq} \,|\, 1 \leq i \leq |\mathcal{F}|,\ 1 \leq j \leq |\mathcal{V}|,\ 1 \leq q \leq K\}. \tag{4.27}$$

$$\Theta^q = \{\theta_k^q \,|\, 1 \leq k \leq |\mathcal{N}|,\ 1 \leq q \leq K\}. \tag{4.28}$$

Here, $\mathcal{X}^q$ is a three-dimensional array, and $\mathcal{Y}^q$ is a four-dimensional array, $\mathcal{H}^q$ is a three-dimensional array, and Θ^q is a two-dimensional array, where a vector for q is added into each array of (4.20), (4.21), (4.23), and (4.24), respectively. Note that $\mathcal{Z}^q$ is not defined because $\mathcal{Z}$ already contains index q.

Algorithm 5: Service Chain Construction Algorithm for Optimization Problem

Input : $\mathcal{F}$

Output : $\mathcal{X}, \mathcal{Y}, \mathcal{Z}, \mathcal{H}, \Theta$

All f^i in $\mathcal{F}$ are sorted in ascending order of $\min l(v_j^i)\bar{l}(v_j^i)$

for $i = 1$ *to* $|\mathcal{F}|$ **do**
> **for** $q = 1$ *to* K **do**
>> **for** *all* $v_j^i \in \mathcal{V}^i$ **do**
>>> **while do**
>>>> $x_j^{iq} \leftarrow x_j^{iq} + \zeta$
>>>> $p_j^i \leftarrow \text{LossCheck}(\frac{t_S^i}{x_j^{iq}},\ \gamma x_j^{iq})$ /*Return the loss probability*/
>>>> **if** $p_j^i \leq P^i$ **then**
>>>>> **break**
>>
>> Security Mgmt. for VNF Sharing($f^i, q, \mathcal{X}^q$) /*Algorithm 6*/
>> Security Mgmt. for New VNF($f^i, q, \mathcal{X}^q$) /*Algorithm 7*/
>> Security Mgmt. for Placement($f^i, q, \mathcal{X}^q$) /*Algorithm 8*/
>> **if** $\sum_{j=1}^{|\mathcal{V}|} \sum_{k=1}^{|\mathcal{N}|} \sum_{q=1}^{K} y_{jk}^{iq} z^{iq} \delta_k^{iq} \neq |\mathcal{V}^i|$ **then**
>>> **for** *all* $v_j^i \in \mathcal{V}^i$ **do**
>>>> $y_{jk}^{iq} \leftarrow 0$
>>>
>>> Security Mgmt. for Placement($f^i, q, \mathcal{X}^q$) /*Algorithm 8*/
>>
>> c_{total}^{iq} is calculated as the objective function value of (4.9) with $\mathcal{X}^q$, $\mathcal{Y}^q$, $\mathcal{Z}$, $\mathcal{H}^q$, and Θ^q
>
> Select index q^* that has the minimum value of c_{total}^{iq}
> **for** *all* $v_j^i \in \mathcal{V}^i$ *and* $n_k \in \mathcal{N}$ **do**
>> $x_j^i \leftarrow x_j^{iq^*}, y_{jk}^i \leftarrow y_{jk}^{iq^*}, z^{iq^*} \leftarrow 1, \eta_j^i \leftarrow \eta_j^{iq^*}, \theta_k \leftarrow \theta_k + \theta_k^{q^*}$
>
> **for** *all* $v_j^i \in \mathcal{V}^i$ **do**
>> **while do**
>>> $x_j^i \leftarrow x_j^i - \zeta$
>>> $p_{jk} \leftarrow \text{LossCheck}(\frac{\sum_{i=1}^{|\mathcal{F}|} t_S^i y_{jk}^i}{\sum_{i=1}^{|\mathcal{F}|} r_{jk}^i},\ \gamma \sum_{i=1}^{|\mathcal{F}|} r_{jk}^i)$
>>> **if** $p_{jk} > P^i$ **then**
>>>> $x_j^i \leftarrow x_j^i + \zeta$
>>>> **break**

All service chains in $\mathcal{F}$ are constructed based on $\mathcal{X}$, $\mathcal{Y}$, $\mathcal{Z}$, $\mathcal{H}$, and Θ

The primary algorithm, Algorithm 5, requires that all service chain requests be built in ascending order of $\min l(v^i_j)\bar{l}(v^i_j)$, where $\min l(v^i_j)$ and $\bar{l}(v^i_j)$ are the minimum and average of the security level of VNFs for f^i, respectively. To minimize the total cost, VNF and node security costs must be considered for constructing a service chain. Here, if the security level of a VNF is increased, it does not affect the placement of other VNFs. However, if the security cost of a node is increased, more VNFs may be placed in the node without incurring additional cost. This means that the security level of a node should be increased rather than that of a VNF. Therefore, in Algorithm 5, service chains are constructed in ascending order in terms of the VNF security level.

For the ith service chain construction, the K shortest paths are checked as a candidate route at lines 5 to 19, and all VNFs in $\mathcal{V}^i$ are processed at lines 6 to 11. Here, as shown in the constraint conditions of the optimization problem, service chains must be constructed so as to satisfy the amount of resources of (4.15) and the loss probability of (4.16). From lines 7 to 11, x^{iq}_j is determined temporarily while increasing its value by ζ, which is the unit resource volume until the loss probability p^i_j becomes smaller than or equal to P^i. This temporal value is appropriate in the worst-case scenario; therefore, it may be reduced after deciding other variables at line 25. Here, the function called $LossCheck(\rho_{jk}, K_{jk})$ returns the loss probability that is calculated from (3.18) to (3.22).

Next, with the three functions called Algorithm 6, 7, and 8, the service chain f^i is constructed by considering the amount of resources and managing the security level of VNFs and nodes from lines 12 to 14. These functions determine the decision variables for f^i on the qth shortest path while minimizing the resource consumption cost and the security cost, which are explained in Sect. 4.3.2. At line 12, VNFs are shared among multiple service chains in Algorithm 6. At line 13, some new VNFs are placed considering the security cost in Algorithm 7. Then, at line 14, most of the rest of VNFs, which have not been placed yet, are placed in Algorithm 8. Some VNFs may not be placed at lines 12 to 14 due to the shortage of available resources or sequence of VNFs, and hence lines 15 checks whether all VNFs have been placed in the nodes. If some VNFs for f^i have not been placed, all VNFs are placed with Algorithm 8 again.

At line 19, the total cost c^{iq}_{total} for f^i on the qth shortest path is calculated as the objective function value (4.9) with $\mathcal{X}^q$, $\mathcal{Y}^q$, $\mathcal{Z}$, $\mathcal{H}^q$, and Θ^q. Next, the q^*th shortest path of f^i, where $c^{iq^*}_{total}$ is the minimum, is determined as the optimal route at line 20. Then, x^i_j, y^i_{jk}, z^{iq}, η^i_j, and θ_k are updated at line 22, and the decision variables for f^i are determined. Here, x^i_j is updated using the temporary value $x^{iq^*}_j$. As previously explained, $x^{iq^*}_j$ may be much larger; therefore, it is reduced at lines 23 to 29 so long (4.16) is satisfied. Finally, all service chains $\mathcal{F}$ are constructed at line 30 using the solutions derived in the above processes.

4.3.2 VNF placement with Security Level Management

Algorithm 6: Security Management for VNF Sharing

1 **Input :** $f^i, q, \mathcal{X}^q$
2 **Output :** $\mathcal{Y}^q$, $\mathcal{H}^q$, Θ^q
3 **for** *all* $v^i_j \in \mathcal{V}^i$ **do**
4   **for** *all* $n_k \in \mathcal{N}$ **do**
5   **if** *v_j exists on n_k, $\sum_{n_k \in \mathcal{N}} y^{iq}_{jk} = 0$, $\delta^{iq}_k = 1$, and Eqs. (4.15), (4.16), and (4.19) are satisfied* **then**
6             $y^{iq}_{jk} \leftarrow 1$
7   **if** $l(v^i_j) < d(n_k)$ **then**
8                 $\eta^{iq}_j \leftarrow d(n_k) - l(v^i_j)$
9   **if** $l(n_k) + \theta_k + \theta^q_k < d(v^i_j)$ **then**
10                 $\theta^q_k \leftarrow d(v^i_j) - l(n_k) - \theta_k$

Our proposed method places VNFs with security-level management in Algorithms 6, 7, and 8. In Algorithm 6, every VNF for f^i is placed at a node preferentially if the VNF has already been placed in the node for one or more service chains between f^1 to f^{i-1}. Here, this VNF placement must consider the amount of resources of (4.15), the loss probability of (4.16), and the sequence of VNFs of (4.19) at line 5. The VNF placement in Algorithm 6 can reduce the resource consumption cost because VNFs are preferentially shared among multiple service chains. The VNF is placed in n_k at line 6, and then η^{iq}_j and θ^q_k is updated at lines 7 to 10. In the heuristic algorithm, unlike the optimization problem, η_j^{iq} and θ_k^q are updated so as to satisfy (4.3) and (4.4), respectively. Note that the security level of n_k is checked with the sum of $l(n_k)$, θ_k, and θ_k^q. Here, θ_k is the amount of increased security level that was determined for the previous service chains, and θ_k^q is its amount determined in the qth shortest path for f^i.

Algorithm 7: Security Management for New VNF

---

1 **Input :** $f^i, q, \mathcal{X}^q$
2 **Output :** $\mathcal{Y}^q$, $\mathcal{H}^q$, Θ^q
3 **for** *all* $v_j^i \in \mathcal{V}^i$ **do**
4     $n_{\dot{k}} \leftarrow -1$
5   **for** *all* $n_k \in \mathcal{N}$ **do**
6   **if** $\sum_{n_k \in \mathcal{N}} y_{jk}^{iq} = 0$, $\delta_k^{iq} = 1$, *and* *Eqs. (4.15), (4.16), and (4.19) are satisfied* **then**
7   **if** $l(n_k) + \theta_k + \theta_k^q \geq \max d(v_j^i)$ *and* $\min l(v_j^i) \geq d(n_k)$ **then**
8                 $n_{\dot{k}} \leftarrow n_k$
9   **break**
10   **else**
11   **if** $\frac{l(n_{\dot{k}})}{d(n_{\dot{k}})R_{\dot{k}}} < \frac{l(n_k)}{d(n_k)R_k}$ **then**
12                     $\frac{l(n_{\dot{k}})}{d(n_{\dot{k}})R_{\dot{k}}} \leftarrow \frac{l(n_k)}{d(n_k)R_k}$
13                     $n_{\dot{k}} \leftarrow n_k$
14   **if** $n_{\dot{k}} \neq -1$ **then**
15         $y_{j,n_{\dot{k}}}^{iq} \leftarrow 1$
16   **if** $l(v_j^i) < d(n_{\dot{k}})$ **then**
17         $\eta_j^{iq} \leftarrow d(n_{\dot{k}}) - l(v_j^i)$
18   **if** $l(n_{\dot{k}}) + \theta_{\dot{k}} + \theta_{\dot{k}}^q < d(v_j^i)$ **then**
19         $\theta_{\dot{k}}^q \leftarrow d(v_j^i) - l(n_{\dot{k}}) - \theta_{\dot{k}}$

Next, in Algorithm 7, some new VNFs for f^i are placed in nodes because they cannot be shared in Algorithm 6. Therefore, the placement of v_j on n_k is not checked at line 6 unlike line 5 of Algorithm 6. For the placement of new VNFs using security-level management, Algorithm 7 considers two cases for the security level of VNFs and nodes. When v_j^i is placed on n_k in the first case at line 7, the security level of the VNF and nodes remain constant. Here, $\max d(v_j^i)$ is the maximum security demand of $v_j^i \in \mathcal{V}^i$, and $\min l(v_j^i)$ is the minimum security level, as previously explained. Not all VNFs for f^i are placed on that node when the node has insufficient resources. In the second case at line 10, the security level of the VNF or nodes must be increased when $v_j^i \in \mathcal{V}^i$ is placed on n_k. To reduce the security cost, the node with a higher security level, lower security demand, and smaller resource is selected as the candidate node n_k by comparing $\frac{l(n_k)}{d(n_k)R_k}$ with $\frac{l(n_k)}{d(n_k)R_k}$ at lines 11 to 13. Thus, the node n_k, where v_j^i should be placed, is updated at lines 7 to 13. From lines 14 and 15, v_j^i is determined to be placed on n_k if n_k is updated at line 8 or 13. Then, η_j^{iq} and θ_k^q are updated at lines 16 to 19 based on (4.3) and (4.4).

Algorithm 8: Security Management for Placement

1 **Input :** $f^i, q, \mathcal{X}^q$
2 **Output :** $\mathcal{Y}^q, \mathcal{H}^q, \Theta^q$
3 **for** *all* $v_j^i \in \mathcal{V}^i$ **do**
4   **for** *all* $n_k \in \mathcal{N}$ **do**
5   **if** $y_{jk}^{iq} = 0$, $\delta_k^{iq} = 1$, *and* *Eqs. (4.15), (4.16), and (4.19) are satisfied* **then**
6             $y_{jk}^{iq} \leftarrow 1$
7   **if** $l(v_j^i) < d(n_k)$ **then**
8                 $\eta_j^{iq} \leftarrow d(n_k) - l(v_j^i)$
9   **if** $l(n_k) + \theta_k + \theta_k^q < d(v_j^i)$ **then**
10                 $\theta_k^q \leftarrow d(v_j^i) - l(n_k) - \theta_k$

Finally, in Algorithm 8, VNFs are placed regardless of the node and VNF security costs At line 6, VNFs are placed in a node where (4.15), (4.16), and (4.19) are satisfied. Then, η_j^{iq} and θ_k^q are updated at lines 7 to 10. Therefore, the security cost increases significantly in Algorithm 8, but any VNFs can always be placed.

4.4 Numerical Examples

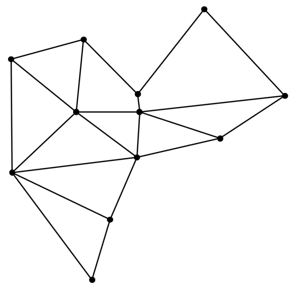

Figure 4.3: Tokyo TMN12.

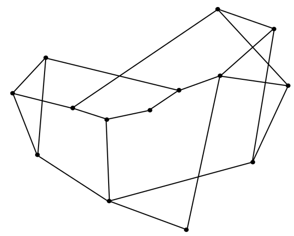

Figure 4.4: NSFnet.

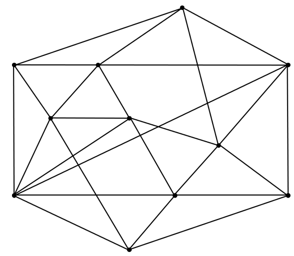

Figure 4.5: COST239.

In this section, we evaluate the performance of our proposed service chain construction with security-level management under three network topologies shown in Figs. 4.3 to 4.5: Tokyo TMN12 [49], NSFnet [48], and COST239 [47]. Tokyo TMN12 is designed with regional railway information for Tokyo metropolitan area, and NSFnet was a backbone network connecting research and educational institutions across the United States. Moreover, COST239 is a network connecting major cities across Europe. The Tokyo TMN12 topology has 12 nodes and 21 links, the NSFnet topology has 14 nodes and 21 links, and the COST239 topology has 11 nodes and 26 links. We assume that each node is a physical node such as a data center. Moreover, the link length is not set to each link, and the transmission route is selected among K shortest paths based on the number of hops. We investigate the performance of our proposed method for three network topologies with a similar number of nodes and links.

In these networks, five types of VNFs are available ($\mathcal{V} = \{v_1, \cdots, v_5\}$), and three VNFs are used in f^i ($|\mathcal{V}^i| = 3$) as is the case with [18, 26, 32, 43]. The three VNFs are randomly selected among the five types of VNFs in $\mathcal{V}$. The maximum amount R_k of resources is 600 or 1,000 randomly, and the bandwidth B_{kl} is set to 1,000 for any pairs of nodes n_k and n_l. When $v_j \in \mathcal{V}$ is placed in a node, a fixed amount $\dot{r}_j$ of resources is needed. Here, $\dot{r}_j$ is set to ten regardless of the VNF type. For the ith service chain f^i, a source node s^i and a destination node d^i are randomly selected among all nodes. Additionally, the amount t_S^i of traffic is equal to 30, and the number K of the shortest paths is set to seven. In terms of the security level and demand, an integer between zero to four is randomly assigned to $l(v^i_j)$, $d(v^i_j)$, $l(n_k)$, and $d(n_k)$, respectively [22, 37]. Note that we set the amount of node resources, the number of service chains, and other parameters so that all requests can be satisfied.

In this scenario, we evaluate the performance of our heuristic-based service chain construction described in Sect. 4.3. Additionally, a method that solves the optimization problem shown in Sect. 4.2 is evaluated using the genetic algorithm (GA). For a performance comparison, the performances of the PLACE and GREEDY methods are also evaluated. The PLACE method only uses Algorithm 8 to manage the security level of the VNFs and nodes for constructing service chains. To reduce the security cost, the GREEDY method attempts to share the VNFs among multiple service chains if (4.1) is satisfied in Algorithm 6. Additionally, in Algorithm 7, the condition at line 10 is replaced as follows: else if $(\min l(v^i_j) \geq d(n_k))$.

4.4.1 Impact of Number of Service Chains

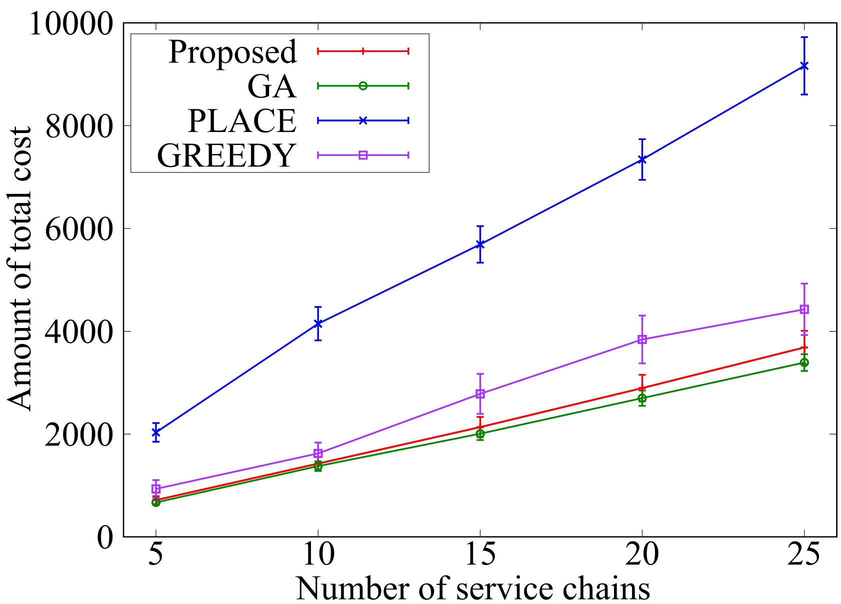

Figure 4.6: Value of objective function vs. number of service chains for Tokyo TMN12.

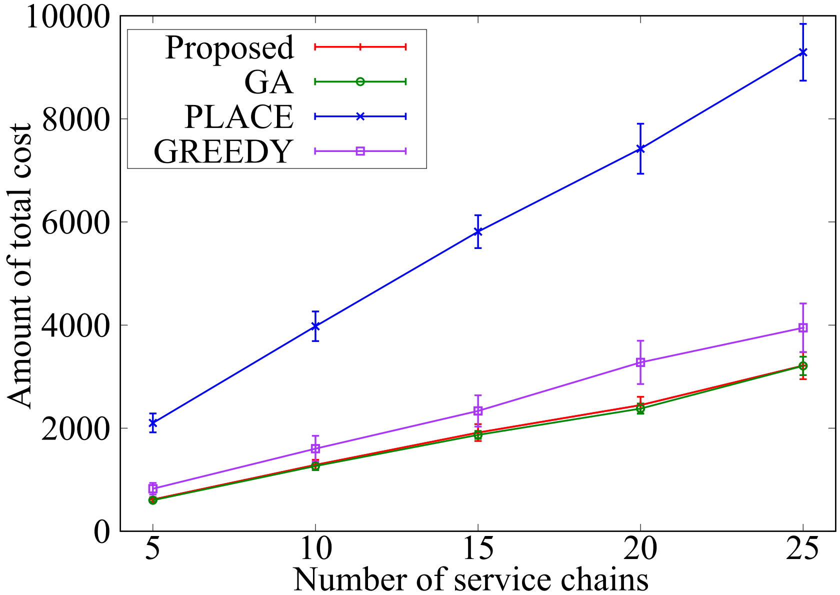

Figure 4.7: Value of objective function vs. number of service chains for NSFnet.

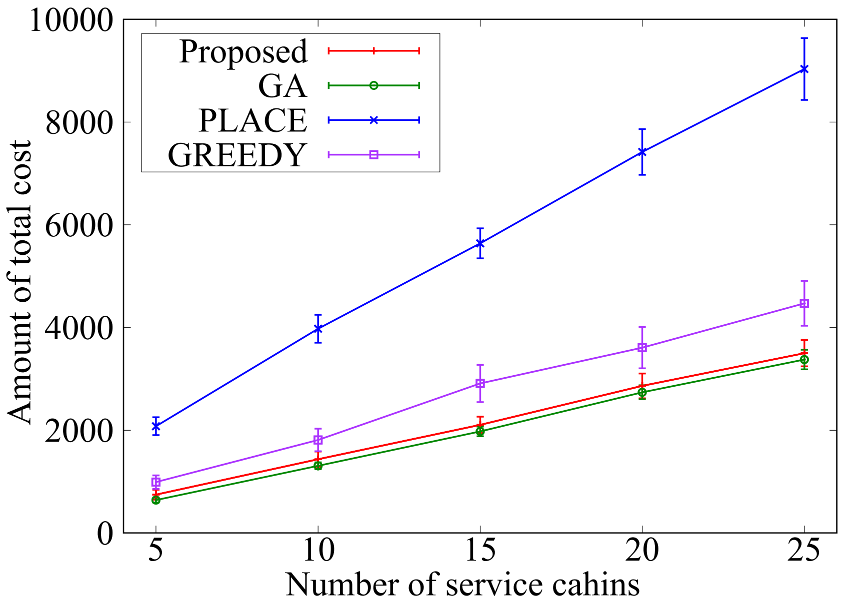

Figure 4.8: Value of objective function vs. number of service chains for COST239.

This section evaluates the impact of the number of service chains on the objective function's value regarding the proposed method's total cost, which is the proposed, GA, PLACE, and GREEDY methods. Here, the parameters α, β, and the upper bound of the loss probability P^i for any f^i are set to 0.1.

Figures 4.6 to 4.8 show how the number of service chains affects the total cost for the four methods in the three topologies. These figures reveal a consistent increase in the total cost for all methods as the number of service chains increases. This is owing to the increased number of placed VNFs, which increases the amount of resources needed to construct all service chains. Moreover, because more VNFs cannot satisfy (4.1), the security cost increases. The total cost of the proposed and GA methods are found to be consistently smaller than the PLACE and GREEDY methods. Additionally, the result of the proposed method is much closer to that of the GA method than the PLACE and GREEDY methods. The findings demonstrate that our proposed method effectively reduces the total cost of service chain construction by adeptly managing the security level of nodes and VNFs.

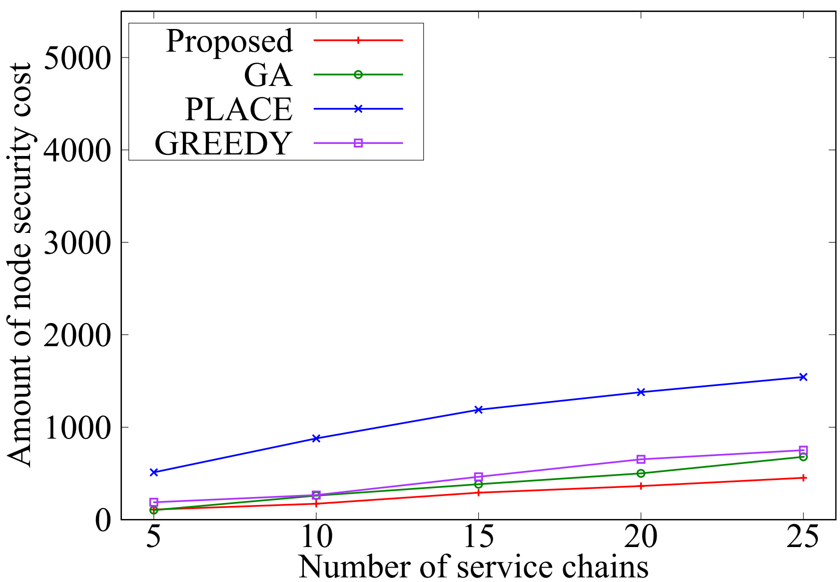

Figure 4.9: Amount of node security cost vs. number of service chains for Tokyo TMN12.

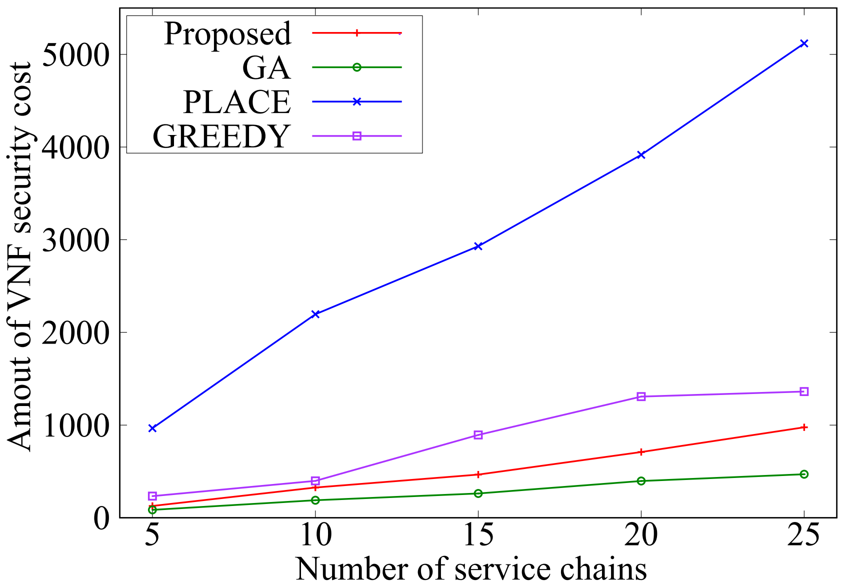

Figure 4.10: Amount of VNF security cost vs. number of service chains for Tokyo TMN12.

Figures 4.9 and 4.10 show how the number of service chains affects the security costs for the four methods for Tokyo TMN12. Here, the VNF and node security costs are expressed as follows: $\sum_{i=1}^{|\mathcal{F}|}\sum_{j=1}^{|\mathcal{V}|} c_{jk}^{i,VNF}$ and $\sum_{k=1}^{|\mathcal{N}|} c_k^{node}$ in (4.9), respectively. These figures demonstrate that the security cost for all methods increases as the number of service chains increases. Additionally, the security costs of the proposed and GA methods are always smaller than those of the PLACE and GREEDY methods. In Fig. 4.9, the amount of node security cost for the proposed method is smaller than that of the GA method. Contrarily, Fig. 4.10 shows that the amount of VNF security cost for the proposed method is greater than that of the GA method.

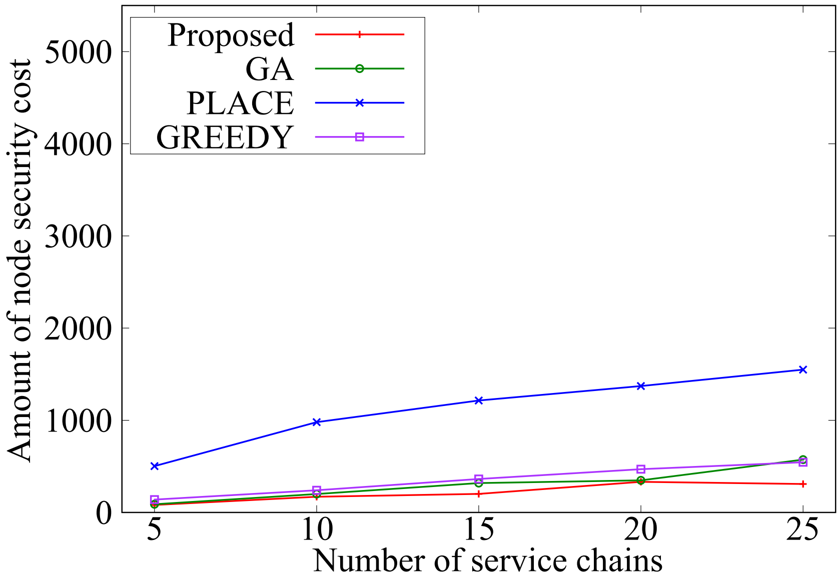

Figure 4.11: Amount of node security cost vs. number of service chains for NSFnet.

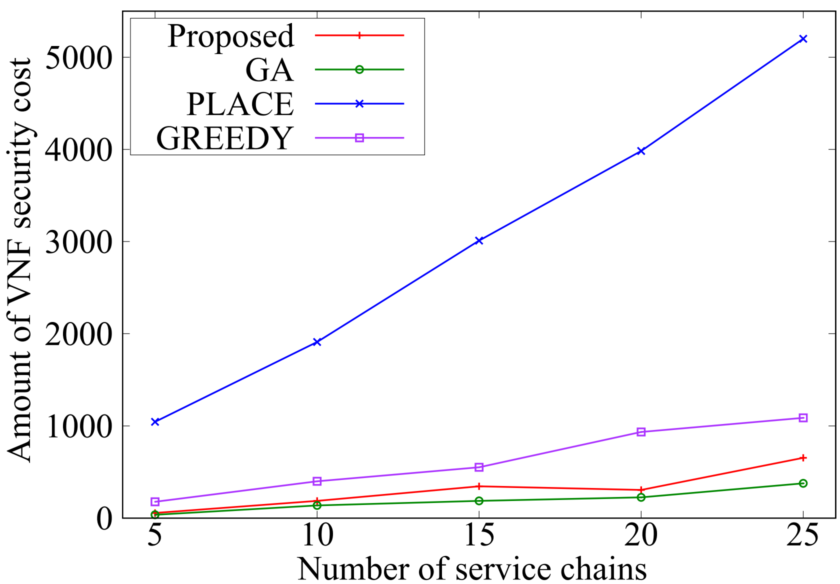

Figure 4.12: Amount of VNF security cost vs. number of service chains for NSFnet.

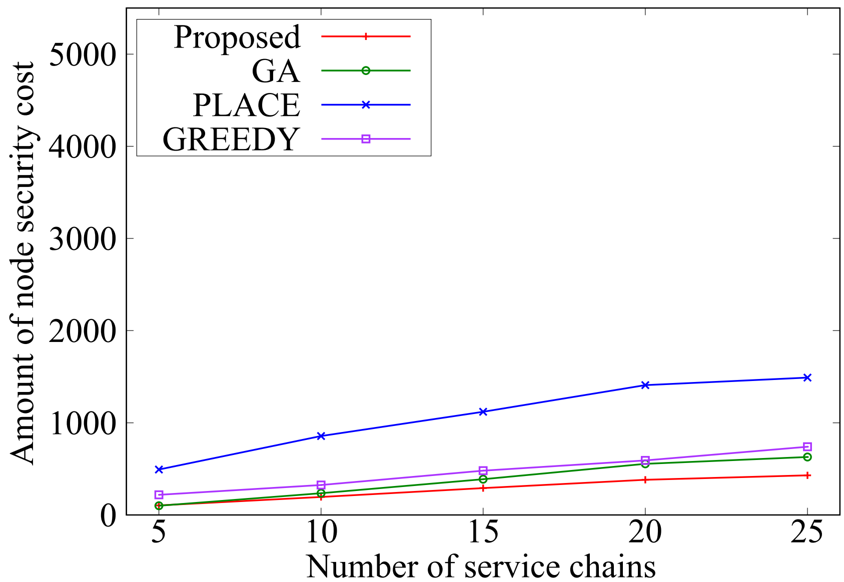

Figure 4.13: Amount of node security cost vs. number of service chains for COST239.

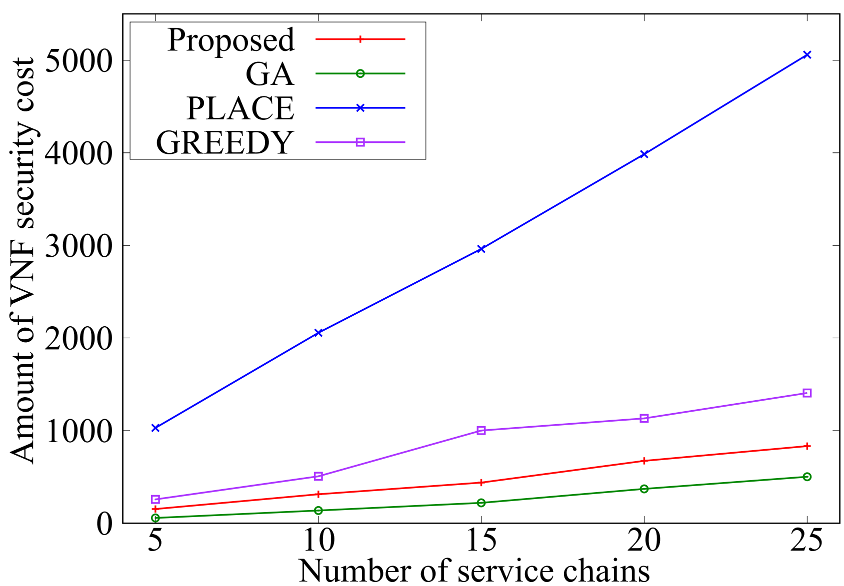

Figure 4.14: Amount of VNF security cost vs. number of service chains for COST239.

Similarly, Figs. 4.11 to 4.14 show how the number of service chains affects the security costs for the four methods in NSFnet and COST239. These figures demonstrate that the amount of node (VNF) security cost for the proposed method is smaller (greater) than that of the GA method, regardless of the topologies.

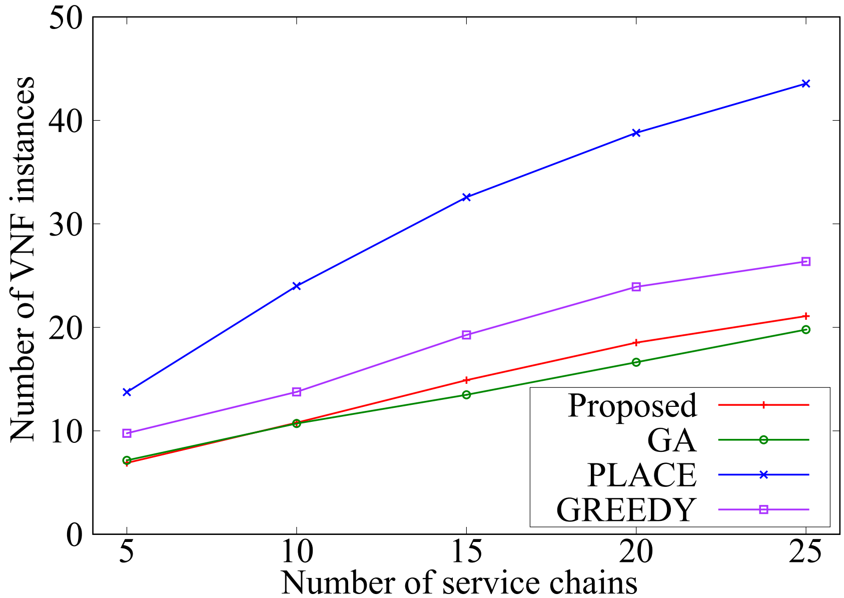

Figure 4.15: Number of VNF instances vs. number of service chains for Tokyo TMN12.

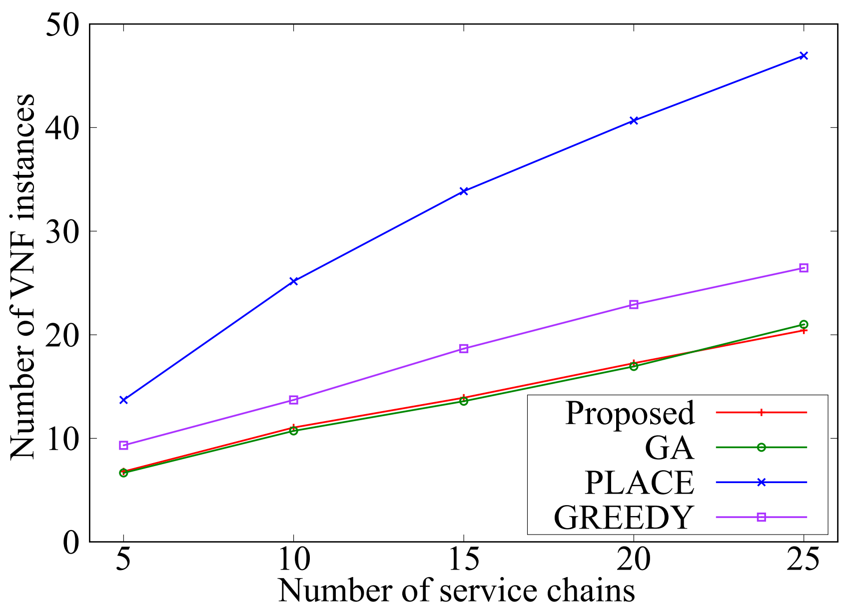

Figure 4.16: Number of VNF instances vs. number of service chains for NSFnet.

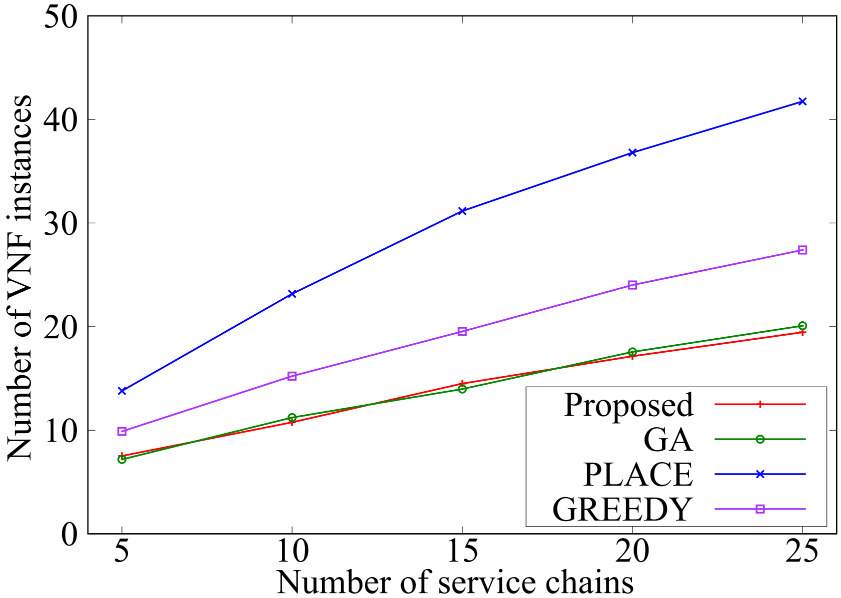

Figure 4.17: Number of VNF instances vs. number of service chains for COST239.

Figures 4.15 to 4.17 show how the number of service chains affects that of VNFs placed for the four methods. These figures demonstrate that the number of VNF instances for all methods increases with an increase in the number of service chains. We find that the number of VNF instances for the proposed and GA methods is consistently smaller than that of the PLACE and GREEDY methods. Moreover, the result of the proposed method is much closer to that of the GA method, regardless of the topologies. In some of these figures, the number of VNFs for the proposed method is slightly lower than that of the GA method. This is because our proposed method shares many VNFs aggressively in Algorithm 6.

These results demonstrate that the proposed and GA methods can significantly reduce the total cost of service chaining by managing the security level and sharing the VNFs among multiple service chains. Additionally, from these figures, there is little to no difference between the proposed and GA methods. In terms of the security cost, we can also confirm that there is no significant difference between the three topologies in Figs. 4.9 to 4.14. This is because the security cost is greatly affected by the values of both the security level and demand for VNFs and nodes. As a result, the impact of network topology becomes small for all the methods. Figures 4.15 to 4.17 also demonstrate that the number of VNF instances remains almost consistent regardless of the topology, as each method endeavors to share VNFs among multiple service chains.

4.4.2 Impact of the VNFs security cost in terms of α

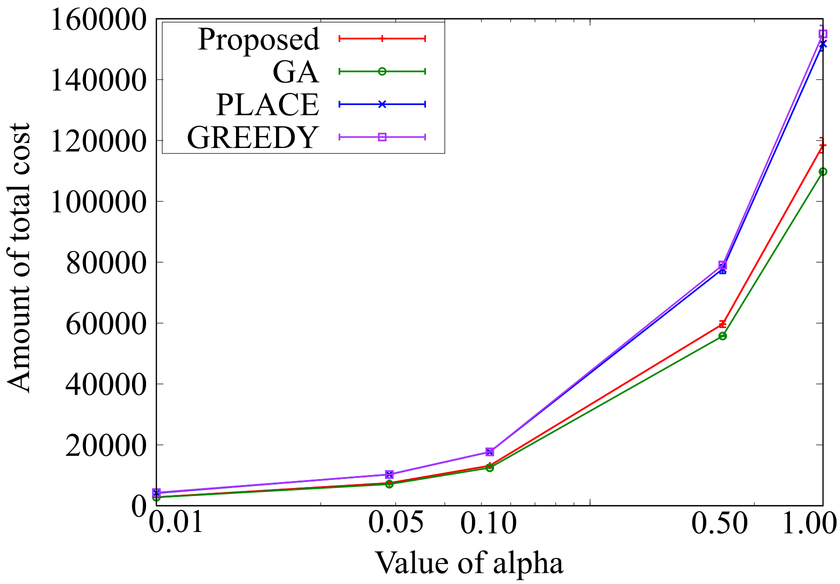

Figure 4.18: Value of objective function vs. value of α for Tokyo TMN12.

This section evaluates the impact of the VNF security cost regarding α on the performance of the four methods. This section only shows the Tokyo TMN12 results, but similar results have been obtained for the other two topologies. Here, the number of service chains is set to 15. Moreover, $l(v_j^i)$ and $d(n_k)$ are set to zero and four, respectively. The $l(n_k)$ and $d(v_j^i)$ are randomly set at zero to four.

Figure 4.18 shows how the VNF security cost affects the total cost of the service chain construction for the four methods. According to the figure, as expected, given an increase in the security cost for each VNF, the costs for all methods increase with an increase in the value of α. We also find that the proposed method's total cost is smaller than that of the PLACE and GREEDY methods even if the value of α increases. Additionally, the proposed's result is closer to that of the GA method. Thus, like the GA method, our proposed method can significantly decrease the total cost regardless of α.

4.4.3 Impact of Amount of Resources for a Node

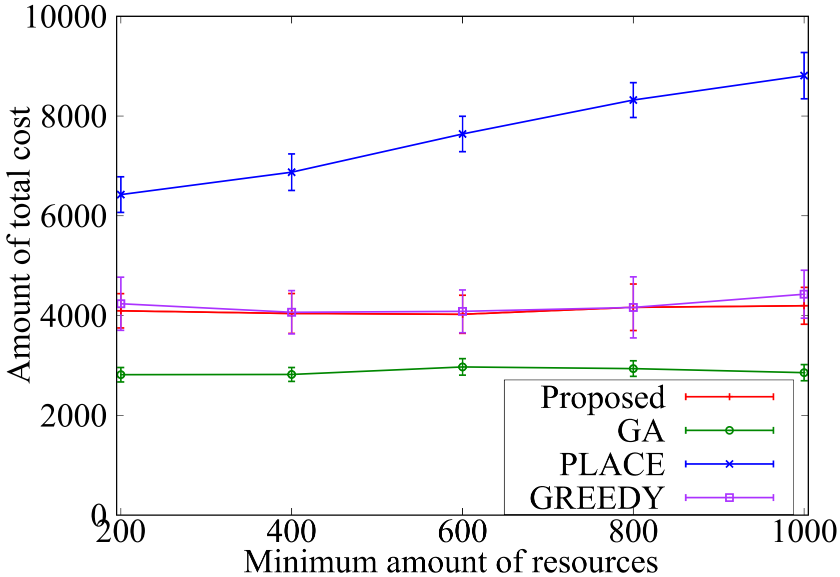

Figure 4.19: Value of objective function vs. value of R_{min} for Tokyo TMN12.

This section evaluates the impact of the minimum amount of resources for a node on the four methods' objective functions. This section also only shows the Tokyo TMN12 results, although similar results have been obtained for the other two topologies. Here, the minimum amount of available resources is represented as R_{min}, and R_k is randomly set to R_{min} or $1,000$. Additionally, $l(n_k)$ and $d(v^i_j)$ are set at zero to four, respectively, to investigate the impact of the node security cost, as described in Sect. 4.4.2. Note that $d(n_k)$ and $l(v^i_j)$ are randomly set at zero to four.

Figure 4.19 shows how the amount of resources for a node affects the four methods' costs. According to this figure, the security costs tend to become large from (4.7) and (4.8) as the minimum amount of resources R_k increases. Additionally, according to this figure, the costs for the three methods, which are the proposed, GA, and GREEDY, are almost constant regardless of the node's amount of resources. Hence, the results demonstrate that all three methods can construct service chains without increasing the security cost regardless of the amount of resources. Additionally, the results demonstrate that the proposed method diverges from the GA method, and that the proposed method is much closer to the GREEDY method.

4.4.4 Computation time of optimal service chain construction

Finally, the calculation time of the proposed, GA, PLACE, and GREEDY methods for the Tokyo TMN12 is evaluated. This evaluation is performed using a computer with macOS Monterey 64-bit, Intel Core i5, and 16 GB memory.

Table 4.4: Calculation time [sec] in Tokyo TMN12.

$\lvert\mathcal{F}\rvert$	5	15	25
Proposed	0.0014	0.0023	0.0032
GA	117	247	859
PLACE	0.0014	0.0021	0.0028
GREEDY	0.0015	0.0023	0.0031

Table 4.4 shows the calculation time of the four methods for Tokyo TMN12. The data in the table indicates that the GA method consistently requires the largest calculation time, irrespective of the number of service chains. However, the proposed, PLACE, and GREEDY methods are significantly smaller compared with the GA method. Except for the GA method, each calculation time is nearly identical, but the PLACE method has the shortest calculation time.

Chapter 5

Conclusion

In this dissertation, we propose two heuristic-based approaches for service chain construction to minimize the cost.

In the first proposed method in Chapter 3, we proposed a cost-effective service chain construction a VNF sharing model with an M/M/1/K queueing model. We evaluated the performance of our proposed method through a simulation, and compared their performances with those of other approaches. Numerical examples showed that our proposed method can construct service chains that minimize the cost for VNFs regardless of the number of service chains, the requirement in terms of the loss probability, the amount of traffic for a service chain, the amount of resources for a node, and the network topologies. In addition, we showed the effectiveness of our proposed method introducing an M/M/1/K queueing model. In particular, we found that our proposed method can minimize the cost for the VNFs by sharing not only the VNF instances but also the resources for processing among multiple service chains.

In the second proposed method in Chapter 4, we proposed a service chain construction with security-level management. The cost optimization problem for constructing the service chains with the security-level management was formulated, and a heuristic algorithm for solving the optimization problem was proposed. Numerical examples demonstrated that our proposed method can construct service chaining to minimize the total security and resource consumption costs. Moreover, we found that our proposed method can significantly decrease the total cost regardless of the three network topologies.

In the future, the proposed method will be extended so that it can be used for network slices.

References

[1] M.D. Ananth and R. Sharma, "Cost and Performance Analysis of Network Function Virtualization Based Cloud Systems," in *Proc. IEEE International Advance Computing Conference (IACC)*, India, Jan. 2017.

[2] R. Mijumbi, J. Serrat, J. Gorricho, N. Bouten, F.D. Turck, and R. Boutaba, "Network Function Virtualization: State-of-the-art and Research Challenges," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 1, pp. 236–262, Sep. 2015.

[3] J. Simarro, R. Moreno-Vozmediano, R. Montero, and I. Llorente, "Dynamic placement of virtual machines for cost optimization in multi-cloud environments," in *Proc. International Conference on High Performance Computing and Simulation (HPCS)*, Turkey, Aug. 2011.

[4] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, April 2010.

[5] C.X. Wang, F. Haider, X. Gao, X.H. You, Y. Yang, D. Yuan, H.M. Aggoune, H. Haas, S. Fletcher, and E. Hepsaydir, "Cellular architecture and key technologies for 5G wireless communication networks," *IEEE Communication Magazine*, vol. 52, no. 2, pp. 122–130, Feb. 2014.

[6] M. Kuzlu, M. Pipattanasomporn, and S. Rahman, "Communication network requirements for major smart grid applications in HAN, NAN and WAN," *Computer Networks*, vol. 67, pp. 74–88, July 2014.

[7] Y. Han, J. Li, D. Hoang, J.-H. Yoo, and J.W.-K. Hong, "An intent-based network virtualization platform for SDN," in *Proc. IEEE 12th International Conference Network and Service Management (CNSM)*, Canada, Nov. 2016.

[8] C.C. Marquezan, L.Z. Granville, G. Nunzi, and M. Burnner, "Distributed autonomic resource management for network virtualization," in *Proc. IEEE Network Operation and Management Symposium (NOMS)*, Japan, April 2010.

[9] Q. Duan, N. Ansari, and M. Toy, "Software-defined network virtualization: an architectural framework for integrating SDN and NFV for service provisioning in future networks," *IEEE Network*, vol. 30, no. 5, pp. 10–16, Sep. 2016.

[10] ETSI - Network Functions Virtualisation (NFV): https://www.etsi.org/technologies/nfv.

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

[12] A. Arulappan, G. Raja, K. Passi, and A Mahanti, " Optimization of 5G/6G telecommunication infrastructure through an NFV-based element management system," *Symmetry*, vol. 14, no. 5, p. 978, May 2022.

[13] F. Schardong, I. Nunes, and A. Schaeffer-Filho, "NFV Resource Allocation: a Systematic Review and Taxonomy of VNF Forwarding Graph Embedding," *Computer Networks*, vol. 185, p. 107726, Feb. 2021.

[14] E. H.-Valencia, S. Izzo, and B. Polonsky, "How will NFV/SDN transform service provider opex?," *IEEE Network*, vol. 29, no. 3, pp. 60–67, June 2015.

[15] D. Bhamare, R. Jain, M. Samaka, and A. Erbad, "A survey on service function chaining," *Journal of Network and Computer Applications*, vol. 75, pp. 138–155, Nov. 2016.

[16] A.M. Medhat, T. Taleb, A. Elmangoush, G.A. Carella, S. Covaci, and T. Magedanz, "Service Function Chaining in Next Generation Networks: State of the Art and Research Challenges," *IEEE Communication Magazine*, vol. 55, no. 2, pp. 216–223, Oct. 2016.

[17] G.L. Santos, D.D.F. Bezerra, E.D.S. Rocha, L. Ferreira, A. Moreira, et al., "Service Function Chain Placement in Distributed Scenarios: a Systematic Review," *Journal of Network and Systems Management*, vol. 30, no. 1, pp. 4:1–4:39, Jan. 2022.

[18] D. Zhai, X. Meng, Z. Yu, and X. Han, "Reliability-Aware Service Function Chain Backup Protection Method," *IEEE Access*, vol. 9, pp. 14660–14676, Jan. 2021.

[19] S.V. Rossem, W. Tavernier, D. Colle, M. Pickavet, and P. Demeester, "Profile-based Resource Allocation for Virtualized Network Functions," *IEEE Transaction on Network and Management*, vol. 16, no.4, pp. 1374–1388, Dec. 2019.

[20] A. Leivadeas, G. Kesidis, M. Ibnkahla, and I. Lambadaris, "Vnf placement optimization at the edge and cloud," *Future Internet*, vol. 11, no. 3, p. 6, Mar. 2019.

[21] J. Pei, P. Hong, K. Xue, D. Li, D.S.L. Wei, and F. Wu, "Two-Phase Virtual Network Function Selection and Chaining Algorithm Based on Deep Learning in SDN/NFV-Enabled Networks," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 6, pp. 1102–1117, June 2020.

[22] S. Liu, Z. Cai, H. Xu, and M. Xu, "Security-aware virtual network embedding," in *Proc. IEEE International Conference on Communications (ICC)*, Sydney, NSW, Australia, pp. 834–840, Jun. 2014.

[23] F. Malandrino, C.F. Chiasserini, G. Einziger, and G. Scalosub, "Reducing Service Deployment Cost Through VNF Sharing," *IEEE/ACM Transactions on Networking*, vol. 27, no. 6, pp. 2363–2376, Dec. 2019.

[24] W. Miao, G. Min, Y. Wu, H. Huang, Z. Zhao, H. Wang, and C. Luo, "Stochastic Performance Analysis of Network Function Virtualization in Future Internet," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 3, pp. 613–626, Mar. 2019.

[25] S. Agarwal, F. Malandrino, C.F. Chiasserini, and S. De, "VNF Placement and Resource Allocation for the Support of Vertical Services in 5G Networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 1, pp. 433–446, Feb. 2019.

[26] D. Amaya, S. Homma, and T. Tachibana, "Service chain construction algorithm for maximizing total data throughput in resource-constrained

NFV environments," *IEICE Trans. on Commun.*, vol. E103-B, no. 4, pp. 335–346, Apr. 2020.

[27] S. Xu, B. Liao, B. Hu, C. Han, C. Yang, Z. Wang, and A. Xiong, "A Reliability-and-Energy-Balanced Service Function Chain Mapping and Migration Method for Internet of Things," *IEEE Access*, vol. 8, pp. 168196–168209, Sept. 2020.

[28] Y. Sumi and T. Tachibana, "Heuristic Service Chain Construction Algorithm Based on VNF Performances for Optimal Data Transmission Services," *IEICE Trans. on Commun.*, vol. E104-B, no. 7, pp. 817–828, July 2021.

[29] K. Tabota and T. Tachibana, "Greedy-Based VNF Placement Algorithm for Dynamic Multipath Service Chaining," *IEICE Trans. on Commun.*, vol. E102-B, no. 3, pp. 429–438, Mar. 2019.

[30] M. Golkarifard, C.F. Chiasserini, F. Malandrino and A. Movaghar, "Dynamic VNF placement, resource allocation and traffic routing in 5G," *Computer Networks*, vol. 188, p. 107830, April 2021.

[31] M. Jia, W. Liang, M. Huang, Z. Xu, and Y. Ma, "Routing Cost Minimization and Throughput Maximization of NFV-Enabled Unicasting in Software-Defined Networks," *IEEE Trans. on Network and Service Management*, vol. 15, no. 2, pp. 732–745, June 2018.

[32] H. Ren, Z. Xu, W. Liang, Q. Xia, P. Zhou, O.F. Rana, A. Galis, and G. Wu, "Efficient Algorithms for Delay-Aware NFV-Enabled Multicasting in Mobile Edge Clouds with Resource Sharing," *IEEE Trans. on Parallel and Distributed Systems*, vol. 31, no. 9, pp. 2050–2066, Sept. 2020.

[33] A.B. Haque, B. Bhushan, and G. Dhiman, "Conceptualizing smart city applications: Requirements, architecture, security issues, and emerging trends," *Expert Systems*, vol. 39, no. 5, p. 12753, Jun. 2022.

[34] M. Uma and G. Padmavathi, "A Survey on Various Cyber Attacks and Their Classification," *International Journal of Network Security*, vol. 15, no. 5, pp. 390–396, Sept. 2013.

[35] F. Reynaud, F. Aguessy, O. Bettan, M. Bouet, and V. Conan, "Attacks against Network Functions Virtualization and Software-Defined Networking: State-of-the-art," in *Proc. Workshop on Security in Virtualized Networks (Sec-Virtnet 2016),*, Jun. 2016.

[36] E. Bou-Harb, M. Debbabi, and C. Assi, "Cyber Scanning: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1496–1519, 20f14.

[37] S. Liu, Z. Cai, H. Xu, and M. Xu, "Towards security-aware virtual network embedding," *Computer Networks*, vol. 91, no. 14, pp. 151–163, Nov. 2015.

[38] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, Fog et al.: A survey and analysis of security threats and challenges Author links open overlay panel," *Future Generation Computer Systems*, vol. 78, no. 2, pp. 680–698, 2018.

[39] C. Basile, F.V. alenza, A. Lioy, D.R. Lopez, and A.P. Perales, "Adding support for automatic enforcement of security policies in NFV networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 2, pp. 707–720, Feb. 2019.

[40] J. Halpern and C. Pignataro, "Service function chaining (SFC) architecture," Internet Eng. Task Force, Fremont, CA, USA, RFC 7665, 2015.

[41] D. Dwiardhika and T. Tachibana, "Optimal Construction of Service Function Chains Based on Security Level for Improving Network Security," *IEEE Access*, vol. 7, pp. 145807–145815, Oct. 2019.

[42] H. Moens and F.D. Turck, "VNF-P: A Model for Efficient Placement of Virtualized Network Functions," in *Proc. 10th International Conference on Network and Service Management (CNSM)*, Nov. 2014.

[43] D. Zhai, X. Meng, Q. Kang, H. Hu, Q. Meng, and Y. Liang, "Security Service Function Chain Deployment Using a Viterbi-Based Algorithm," in *Proc. 2021 13th International Conference on Communication Software and Networks (ICCSN)*, June 2021.

[44] D. Zhai, X. Meng, Z. Yu, H. Hu, and T. Huang, "A Security-aware Service Function Chain Deployment Method for Load Balance and Delay Optimization," *Scientific Reports*, vol. 12, 10442, Oct. 2022.

[45] A.M. Zarca, M. Bagaa, J.B. Bernabe, T. Taleb, and A.F. Skarmeta, "Semantic-Aware Security Orchestration in SDN/NFV-Enabled IoT Systems," *Sensors*, vol. 20, no. 13, p. 3622, Jun. 2020.

[46] T. Sakano, Y. Tsukishima, H. Hasegawa, T. Tsuritani, Y. Hirota, S. Arakawa, and H. Tode, "A study on a photonic network model based on the regional characteristics of Japan," *IEICE Techical Report (PN-2013-1)*, vol. 113, no. 91, pp. 1–6, June 2013 (in Japanese).

[47] M.J. O'Mahony, "Results from the COST 239 project. Ultra-High Capacity Optical Transmission Networks," in *Proc. European Conference on Optical Communication*, pp. 11–14, Sept. 1996.

[48] K.C. Claffy, G.C. Polyzos, H.-W. Braun, "Traffic characteristics of the T1 NSFNET backbone," in *Proc. IEEE INFOCOM '93*, pp. 885–892, Mar. 1993.

[49] T. Tachibana, Y. Hirota, K. Suzuki, T. Tsuritani, and H. Hasegawa, "Metropolitan Area Network Model Design using Regional Railways Information for Beyond 5G Research," *IEICE TRANSACTIONS on Communications*, vol. E106-B, 2023.