Service Function Chain Optimal Composition and Embedding: A Systematic Survey of Optimisation Approaches

Theviyanthan Krishnamohan

School of Computing Science, University of Glasgow, Glasgow, United Kingdom

Paul Harvey

School of Computing Science, University of Glasgow, Glasgow, United Kingdom

Abstract

Service Function Chains (SFCs) support computer networks to keep pace with increasing user numbers and changing usage patterns by enabling network programmability. SFCs virtualise network functions, such as firewalls, enabling them to be programmatically embedded on servers and linked, creating a chain of virtual network functions. Optimally composing and embedding SFCs on physical networks is an $\mathcal{NP}$-hard optimisation problem. Since 2016, there has not been a comprehensive literature survey of approaches to optimally compose and embed SFCs across all application domains.

In this work, we survey the literature, identify, and analyse 209 papers based on their title, abstract and content. We then develop an analytical framework to extract data from the curated papers. Based on the data, we propose an update to the existing three-stage definition of the optimal SFC composition and embedding problem, factoring in recent technological advancements and approaches. We analyse the extracted data in terms of use cases, including optimisation objectives, application domains, physical network topologies, algorithms used, their scalability, their adaptability to a dynamic network environment, and the evaluation mechanisms used. Based on this analysis, we finally present emerging trends and identify research gaps in the literature.

Keywords: Network Function Virtualisation, Software Defined Networking, Service Function Chaining, Network Optimisation, Virtual Network Function

Email addresses: `theviyanthan.krishnamohan@glasgow.ac.uk` (Theviyanthan Krishnamohan), `paul.harvey@glasgow.ac.uk` (Paul Harvey)

1. Introduction

Computer networks are a fundamental part of modern society, playing a pivotal role in healthcare [1], finance [2], governance [3] and entertainment [4]. This makes their safe and reliable operation essential to the smooth functioning of our society, requiring them to keep up with the changing ways in which we use them and the increase in user demand. Yet, traditional networks require manual intervention to adapt to changes in user demand and the operating environment, compromising on safety and reliability. Network programmability, by enabling programmatic control of networks, enables faster adaptation to these changes, ensuring safety and reliability [5].

Service Function Chaining (SFC) [6] is one way of realising network programmability. As shown in Fig. 1, SFC uses Network Function Virtualisation (NFV) [7] to virtualise network functions, such as a firewall, so that they can be programmatically embedded on servers, and uses Software Defined Networking (SDN) [8] to programmatically route traffic between these Virtual Network Functions (VNFs), creating a chain of VNFs. Optimally composing and embedding SFCs on a physical network is a challenging problem, requiring an optimisation approach [9].

Herrera et al. [9] surveyed approaches to optimally compose and embed SFCs in 2016 and defined the optimal SFC composition and embedding problem in terms of three stages. Since then, to the best of our knowledge, 5 papers [10–14] have surveyed this landscape; however, they surveyed approaches to optimising only one of the sub-problems [11, 12], focused only on a specific application scenario [12–14], such as 5G or distributed environments, or did not consider the definition introduced by Herrera et al. [10, 13, 14].

In this paper, we systematically survey 209 papers published between 2014 and 2025, and critically analyse the three-stage problem definition introduced by Herrera et al. [9] on the backdrop of recent advancements and approaches, and introduce an updated three-stage problem definition (Section 4). We analyse the papers in terms of their use cases (Section 5), such as optimisation objectives, application domains, and network topologies, algorithms used (Section 7), their scalability and dynamic nature, and evaluation mechanisms (Section 8). We, then, identify and discuss emerging trends (Section 9) in approaches and gaps in existing research (Section 10). We believe this paper will help the research community by providing the state of the art of optimal SFC composition and embedding approaches, and guide future research by identifying research gaps and emerging trends in the literature. Table 1 lists the acronyms used in this paper and their definition.

2. Background

We now provide an overview of SFC, its optimisation challenges, and contextualise related surveys.

Acronym	Definition
SFC	Service Function Chain
NFV	Network Function Virtualisation
SDN	Software Defined Networking
VNF	Virtual Network Function
VL	Virtual Link
ETSI	European Telecommunication Standards Institute
IETF	Internet Engineering Task Force
NFV-RA	Network Function Virtualisation-Resource Allocation
VNF-CC	Virtual Network Function-Chain Composition
VNF-FGE	Virtual Network Function-Forwarding Graph Embedding
VNF-SCH	Virtual Network Function-Scheduling
SFCR	Service Function Chain Request
VNF-FG	Virtual Network Function-Forwarding Graph
SFC-OCE	Service Function Chain-Optimal Composition and Embedding
VNF-EM	Virtual Network Function-Embedding
VL-EM	Virtual Link-Embedding
ISP	Internet Service Provider
IoT	Internet of Things
WDM	Wavelength Division Multiplexing
MEC	Multi-access Edge Computing
EON	Elastic Optical Network
A3C	Asynchronous Advantage Actor-Critic-based
RAN	Radio Access Network
RL	Reinforcement Learning
NN	Neural Network
LSTM	Long Short-Term Memory
MDP	Markov Decision Process
RNN	Recurrent Neural Network
DQN	Deep Q-Network
ML	Machine Learning
GA	Genetic Algorithm
GWO	Grey Wolf Optimisation
PGVT	Production-Grade Virtualisation Tools
DT	Digital Twin

Table 1: Acronyms used in this paper and their definitions

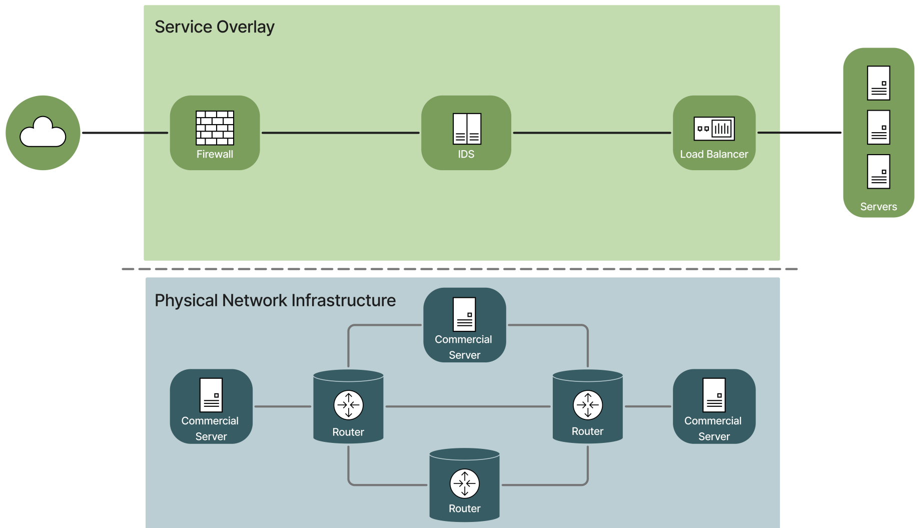

Figure 1: SFC creates a virtual service overlay over the physical network by using NFV and SDN.

2.1. Service Function Chaining

In traditional networks, network functions, such as firewalls, are hardware devices [6]. These hardware devices require significant capital expenditure [14], due to the cost of hardware [14], and operational expenditure, as hardware devices consume power and physical space, while requiring manual intervention to reconfigure networks [15]. As computer networks become increasingly complex and the role of computer networks expands, these costs become prohibitive, and the need for manual intervention hinders faster adaptation to changes in the operational environment [5].

SFC addresses these issues by combining both NFV and SDN [6]. NFV virtualises network functions, separating the software from hardware, allowing them to be embedded on any server in a network's topology programmatically [7]. These virtualised network functions, called VNFs, offer several benefits over traditional network functions, such as reduced capital expenditure, as VNFs incur only software licensing costs [16], reduced operational expenditure, as multiple VNFs can be embedded on a server [17], and faster adaptation to changes in user demand, as VNFs can be moved from server to server programmatically [17]. Programmatic embedding of VNFs is complemented by programmatic routing of traffic between VNFs using SDN. This effectively creates a virtual service overlay consisting of VNFs and Virtual Links (VLs) over the physical network [6], as shown in Fig. 1.

The European Telecommunication Standards Institute (ETSI) proposed an NFV architectural framework [7] for the management and orchestration of VNFs in the physical network. This deals with embedding VNFs at appropriate network locations, scaling hardware resources, keeping track of where VNFs are embedded, and fault detection and recovery. The Internet Engineering Task Force (IETF) proposed an SFC architecture [18] to steer traffic via VNFs in a specified order. This architecture consists of an SFC encapsulation that associates user traffic with an SFC, the Service Classification Function that routes user traffic through the appropriate SFC based on the encapsulation and Service Function Forwarders that direct traffic to one of their associated VNFs according to the encapsulation.

2.2. NFV-Resource Allocation

SFCs, which include VNFs and VLs, have to be embedded (deployed) on the physical network such that the business goals of a service provider, such as maximising revenue and satisfying Service Level Agreements, are met. Herrera et al. [9] defined this as the *NFV-Resource Allocation* (NFV-RA) problem and broke this down to three stages, each consisting of a sub-problem, namely the VNF-Chain Composition (VNF-CC) sub-problem, the VNF-Forwarding Graph Embedding (VNF-FGE) sub-problem, and the VNF-Scheduling (VNF-SCH) sub-problem. NFV-RA problem is an $\mathcal{NP}$-hard optimisation problem [9].

2.2.1. VNF-CC

VNF-CC deals with the ordering of VNFs in an SFC to optimise the NFV-RA problem. An SFC Request (SFCR), which is a request to deploy an SFC on the physical network, may specify the order between some VNFs but not all, in which case the service provider has the flexibility of deciding the order of the rest of the VNFs. Depending on the order of the VNFs, the resource requirements of the VNFs, such as the required number of CPUs and bandwidth requirements, could vary [9]. Accordingly, there exists an order of VNFs in an SFC that produces an optimal solution to the NFV-RA problem when trying to optimise an objective. The output of the VNF-CC problem is the VNF-Forwarding Graph (VNF-FG), which specifies the order of VNFs in an SFC.

2.2.2. VNF-FGE

VNF-FGE deals with the embedding of VNF-FG on the physical network to optimise the NFV-RA problem. This involves embedding the VNFs on servers and embedding the VLs on physical links. The number and type of VNFs embedded on a server and the amount of traffic flowing via the physical links can impact the performance of the SFC [9]. So, for a given optimisation objective, the goal of VNF-FGE is to find an embedding scheme that produces an optimal solution to the NFV-RA problem.

2.2.3. VNF-SCH

VNF-SCH deals with scheduling VNFs on a server to optimise the NFV-RA problem, assuming only one VNF can be executed on the server at a given time. The execution time of a VNF is dependent on the resources available in a server, and as a result, varies from server to server. Within a server, the VNFs can be executed in any order. These two factors decide the total execution time of an

SFC. VNF-SCH is concerned with finding an execution order for VNFs in each server to optimise the NFV-RA problem.

2.2.4. Coordination of NFV-RA Sub-problems

Based on how approaches optimise the NFV-RA sub-problems, they can be classified as *coordinated* or *uncoordinated*. An uncoordinated approach optimises the VNF-CC problem first and uses its output as an input to the VNF-FGE problem, and so on. In contrast to this sequential approach, a coordinated approach solves the sub-problems simultaneously, such that the NFV-RA sub-problems are collectively optimised. An uncoordinated approach may produce a sub-optimal solution because the optimal solution to each sub-problem may not necessarily produce an optimal solution to the NFV-RA problem overall [9]. Sometimes, a sub-optimal solution to one sub-problem may produce an optimal solution overall. Consequently, an optimal approach should involve optimising all three sub-problems simultaneously.

2.3. Related Surveys

We found 6 papers that surveyed SFC composition and embedding approaches. Herrera et al. [9] surveyed the SFC composition and embedding approaches in 2016 and defined the optimal SFC composition and embedding problem as the NFV-RA problem, involving three sub-problems. Kaur et al. [10] performed a survey of SFC provisioning approaches in 2020. However, this survey did not examine the existing literature in terms of the NFV-RA definition introduced by Herrera et al., and as a result, did not detail what sub-problems the approaches optimised and how they did it. In contrast, Schardong et al. [11] performed a systematic review of SFC embedding approaches based on the NFV-RA definition in 2021, but the review included only approaches that optimised the VNF-FGE sub-problem. Zhang et al. [12] also reviewed the literature on VNF-FGE optimisation approaches in 2021, but the scope was restricted to 5G and 6G networks. Santos et al. [14] reviewed SFC embedding approaches in distributed scenarios in 2022, but the review was not based on NFV-RA. Attaoui et al. [13] surveyed VNF embedding approaches in 5G networks in 2023 without considering NFV-RA.

In summary, surveys since Herrera et al.'s seminal survey on SFC composition and embedding approaches have not surveyed approaches to all NFV-RA sub-problems, restricted the scope to a specific scenario, such as 6G, or did not consider NFV-RA. Table 2 provides a summary of the surveys on SFC composition and embedding approaches. Thus, it can be seen that there exists a need to survey recent literature on all three NFV-RA sub-problems across wider application domains to get an understanding of the recent approaches, emerging trends and research gaps.

3. Methodology

We used Google Scholar to find relevant literature from 2014 to 2025 by running separate searches for each of these search terms: "NFV-RA", "NFV-

Survey	Scope	Considers NFV-RA	NFV-RA Sub-Problems	Year
[9]	All	✓	All	2016
[10]	All	X	N/A	2020
[11]	All	✓	VNF-FGE	2021
[12]	B5G/6G	✓	VNF-FGE	2021
[14]	Distributed Scenarios	X	N/A	2022
[13]	5G	X	N/A	2023

Table 2: A summary of the surveys on SFC embedding approaches

Resource Allocation", "VNF-CC", "VNF-Chain Composition", "VNF-FGE", "VNF-Forwarding Graph Embedding", "VNF-SCH", and "VNF-Scheduling". We performed these searches starting from August 2023 to January 2026. We discovered a total of 2,837 papers, as shown in Table 3, which provides the number of papers returned by each query. We first removed duplicates by considering the title and author list. We then excluded irrelevant papers using the criteria below and examined the abstracts of the remaining papers to select the relevant ones for a full review of the text. We then reviewed the references of the papers selected from the full-text review by repeating the above steps, as shown in Fig. 3. The exclusion criteria are:

- the paper was not peer reviewed
- the paper was not written in English
- the paper was not published between 2014 and 2025
- the paper does not address optimal SFC composition or embedding
- the paper focuses only on VNF embedding outwith the context of SFC embedding
- the paper focuses only on Virtual Network Embedding outwith the context of SFC embedding
- the paper does not propose an approach to SFC composition and embedding

This produced 209 papers from 2014 to 2025.

The majority of these 209 papers were published by the Institute of Electrical and Electronics Engineers (IEEE), as shown in Fig. 2. The rest of the papers were published by Springer, Elsevier, the Association for Computing Machinery (ACM), Wiley and Multidisciplinary Digital Publishing Institute (MDPI).

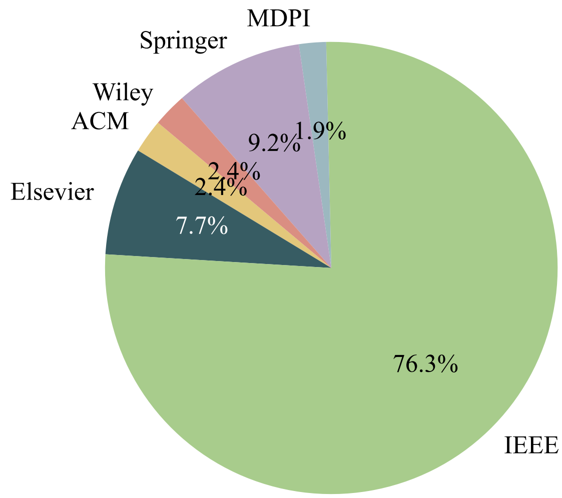

Figure 2: The percentage of surveyed papers from each publisher

Query	**No. of Papers**
NFV-RA	179
NFV-Resource Allocation	721
VNF-CC	30
VNF-Chain Composition	373
VNF-FGE	117
VNF-Forwarding Graph Embedding	907
VNF-SCH	34
VNF-Scheduling	476

Table 3: Number of papers returned for each search query

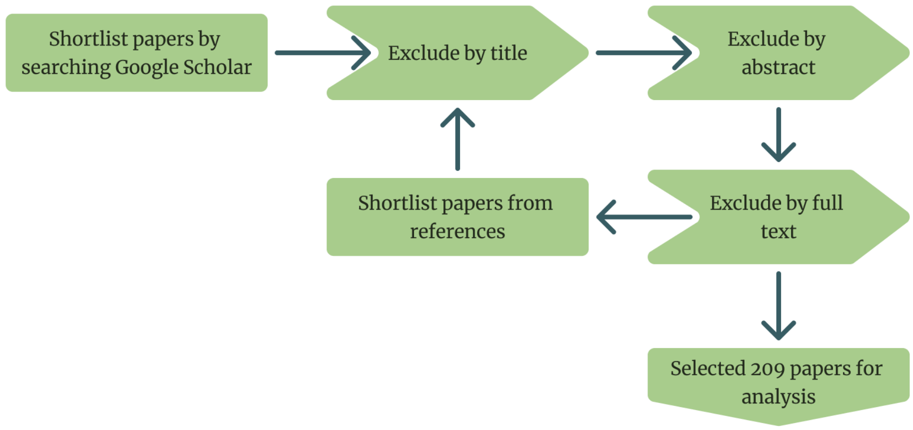

Figure 3: A flow diagram demonstrating the survey methodology.

We then developed an analytical framework to extract useful data for analysis from the selected papers. The analytical framework consisted of a) the year of publication, b) the optimisation objectives, c) the application domains, d) the network topologies used, e) the NFV-RA sub-problems optimised, f) the scalability and dynamic nature of the approaches, and g) the evaluation mechanisms. We then analysed the 209 papers in detail to extract data according to this analytical framework, and used the extracted data to analyse the research landscape between 2014 and 2025.

4. SFC-OCE: an Update to NFV-RA

Following the analysis of the 209 papers, we found that the NFV-RA definition introduced by Herrera et al. [9] in 2016 required an update to accommodate recent progress in virtualisation technologies and the emerging trends in SFC composition and embedding approaches. We propose splitting the VNF-FGE sub-problem into two sub-problems, while disregarding the VNF-SCH sub-problem entirely.

4.1. VNF-FGE

The VNF-FGE sub-problem focuses on embedding the VNFs and the VLs on the physical network. Accordingly, it involves two steps: embedding the VNFs on servers, and embedding the VLs on physical links. However, VNF-FGE, by combining these steps, fails to provide an accurate account of the SFC embedding approaches. The analysis of the 209 papers revealed that 26.3% of the papers that dealt with the VNF-FGE sub-problem did not consider the VL embedding step (Section 6.1), focusing exclusively on the VNF embedding problem. Therefore, there exists a need to separately classify these two sub-problems of VNF-FGE to more accurately classify the recent approaches.

4.2. VNF-SCH

The VNF-SCH sub-problem considers scheduling VNFs in a server such that the total execution time of an SFC is minimised [9]. This is based on the assumption that only one VNF can be run on a server at a given time; however, no justification has been provided for this assumption [9, 19–23]. This assumption has been disputed in the literature [24] as the advancement in virtualisation technologies, such as containerisation, enables multiple VNFs to be executed concurrently [13, 25, 26]. Furthermore, the fact that the 21 papers that optimised the VNF-SCH sub-problem performed only numerical evaluation or simplified simulation instead of using an emulator or a physical testbed, which requires implementation of their solution, to evaluate their approaches (Section 8) amplifies the scepticism on the practical relevance of this sub-problem. Consequently, we decided to omit this sub-problem from the updated definition. However, it is worth mentioning that, given the possibility of implementing VNFs on Field-Programmable Gate Arrays (FPGAs) [27, 28], and the difficulty of executing VNFs concurrently on some resource-constrained edge servers, VNF-SCH may potentially become relevant in the future.

4.3. SFC-OCE

Thus, we now present an updated definition of the SFC embedding problem called the **SFC-Optimal Composition and Embedding (SFC-OCE)** problem. We provide a new name for this definition for two reasons: first, the name NFV-RA refers only to VNF embedding, ignoring VL embedding, and second, to distinguish the updated definition from the established definition to avoid confusion. The SFC-OCE problem consists of three stages, each a sub-problem, namely VNF-CC, VNF-Embedding (VNF-EM), and VL-Embedding (VL-EM). The VNF-CC sub-problem is the same as its namesake in the NFV-RA definition. The VNF-FGE sub-problem in NFV-RA is split into VNF-EM, which deals with embedding VNFs on servers, and VL-EM, which deals with embedding VLs on physical links. VNF-SCH has been dropped due to its practical irrelevance.

4.4. Simultaneous vs. Sequential Approach

Herrera et al. [9] classified NFV-RA approaches into coordinated and uncoordinated (see Section 2.2.4) based on how they optimised multiple NFV-RA sub-problems. In the SFC-OCE definition, we refer to coordinated strategies as simultaneous and uncoordinated strategies as sequential, as we believe these terms are more self-explanatory. In a sequential strategy, VNF-CC, VNF-EM and VL-EM are optimised sequentially, i.e., the output of VNF-CC becomes the input to VNF-EM, and the output of VNF-EM becomes the input to VL-EM. In contrast, in a simultaneous strategy, all three sub-problems are optimised simultaneously. Similar to the NFV-RA problem, a simultaneous strategy is more likely to produce an optimal solution than a sequential strategy [9].

Optimisation Objective	Studies
Cost	[20, 24, 29–95]
Latency	[19–24, 29–31, 49, 61, 62, 67–74, 82, 86, 88, 96–141]
Resource Usage	[24, 65, 66, 73, 84, 86–88, 100–102, 106, 108, 111–113, 121, 125, 127, 128, 133, 136, 139, 141–169, 169–185]
Acceptance Ratio	[20, 57, 70, 85, 116, 137, 140, 142, 143, 168, 169, 175, 177, 179, 180, 182, 183, 186–203]
Power Consumption	[61, 64, 65, 75, 87, 89, 94, 120, 144, 196, 202, 204–209]
Throughput	[67, 100, 108, 125, 210]
Revenue	[76–78, 81, 198, 199, 211, 212]
Reconfiguration	[29, 46, 63, 85, 137, 147, 209, 211, 213]
Number of VNF Instances	[15, 214, 215]
Number of Hosts	[127, 147, 173, 197, 216–218]
Load Balancing	[143, 177, 214]
Traffic Congestion	[59, 60, 116, 139, 218, 219]
Availability	[58, 68, 144, 195, 204, 213, 220]

Table 4: Classification of studies based on their optimisation objectives.

5. SFC-OCE Use Cases

Given the SFC-OCE problem space, we now analyse the use cases of the SFC-OCE approaches in the literature. The use cases include the optimisation objectives, the application domains, and the network topologies. An analysis of the use cases in the literature is important to identify how and where SFCs are being used.

5.1. Categorisation of Optimisation Objectives

The SFC-OCE problem is an optimisation problem in which the SFCs have to be composed and embedded on the physical network, such that certain objectives are optimised. The optimisation objectives are dependent on the business requirements of a network service provider, and hence, can vary from one provider to another. During the analysis of the optimisation objectives in the literature, we found similar objectives being referred to by different terms. For instance, delay [73, 105, 116, 117, 120], and latency [61, 97, 98, 100, 102, 104, 109] were used to refer to the amount of time taken by traffic to traverse an SFC. We also found that some objectives were special cases of generalizable cases. For instance, CPU usage [142, 148, 163, 165, 168, 178, 180] and link usage [144, 145, 151, 159, 162, 165] can be generalised to resource usage. On

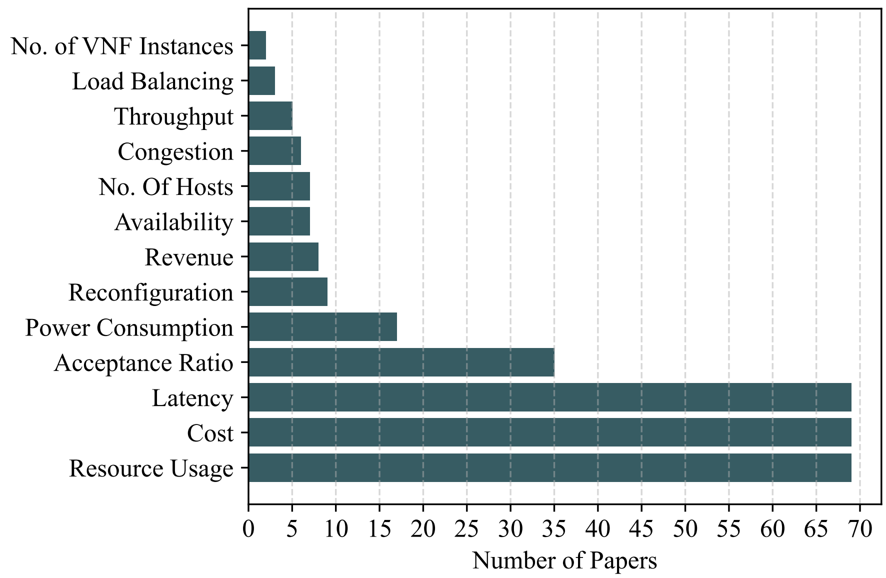

Figure 4: Optimisation objectives used by SFC-OCE approaches.

that account, we introduce a categorisation that generalises and categorises the optimisation objectives into 13 objectives. Fig. 4 shows the number of studies optimising each objective, while Table 4 shows the studies optimising each objective. Some studies considered multiple optimisation objectives. We classified such studies under each of their optimisation objectives. For instance, Araujo et al. [199] optimised both revenue and acceptance ratio, so we classified this study under both revenue and acceptance ratio.

5.1.1. Cost

Cost was the joint most popular optimisation objective in the literature, minimised by 69 studies. This objective considered the monetary cost associated with the utilisation of different resources, such as CPU [20, 38], and link [24, 50], VNF licenses [33, 84], operation [54, 59, 60], and capital [24, 35, 84].

5.1.2. Latency

Latency is the time taken by traffic to traverse an SFC and was referred to by other different terms in the literature, such as delay [73, 105, 116, 117, 120], and execution time [19, 22, 23, 116, 121–124], a term used by studies that optimised the VNF-SCH sub-problem of NFV-RA. 69 studies optimised latency, making it the joint most popular optimisation objective.

5.1.3. Resource Usage

Resource usage was the joint most popular optimisation objective, with 69 studies considering it in the literature. This objective is a generalisation of various types of resource usage found in the literature, such as CPU and link usage. Even though most studies aimed to minimise the resource usage, some studies aimed to maximise it [111, 141, 149, 172], with the rationale being that available network and computing resources should be maximised.

5.1.4. Acceptance Ratio

Acceptance ratio is the number of SFCs that can be accepted to be embedded divided by the total number of SFCRs a network service provider receives, which was maximised by 35 studies in the literature. Maximising the acceptance ratio is predicated upon the efficient use of the resources available in the network, and a higher acceptance ratio often comes at the cost of performance degradation [140].

5.1.5. Power Consumption

Power consumption is the amount of power consumed by the physical network hosting the SFCs. 17 studies used this objective with the goal of minimising it. Power consumption was commonly defined in terms of resources utilised [206, 207], and thus can be considered very similar to the resource usage and cost objectives.

5.1.6. Throughput

Throughput is the rate at which an SFC processes data flowing through it. Maximising throughput was used as an objective in 5 studies.

5.1.7. Revenue

Revenue is the monetary benefit a service provider earns from clients by embedding their SFCRs. It is another objective that was defined in terms of resource utilisation and was maximised by 8 studies.

5.1.8. Reconfiguration

Reconfiguration was used as an optimisation objective in SFC-OCE approaches that considered dynamic network environments (Section 7.3). The dynamic environments considered included environments where SFCRs continued to arrive, and the traffic to SFCs varied dynamically. Often, embedding SFCs when SFCRs continue to arrive, or the traffic continues to vary, involves reconfiguring the existing SFC embedding, which includes migrating VNFs and VLs. Accordingly, the reconfiguration objective involved minimising these migrations, which could lead to service disruptions. 9 studies in the literature minimised reconfiguration.

5.1.9. *Number of VNF Instances*

3 studies aimed to minimise the number of VNF instances to reduce operator costs. Luizelli et al. [15] considered the licensing costs of VNF instances, whereas Bagaa et al. [214] aimed to minimise the number of virtual instances of the packet data network gateway in mobile networks.

5.1.10. *Number of Hosts*

7 studies in the literature aimed to minimise the number of hosts used to embed VNFs. Reducing the number of hosts used to embed VNFs minimises the number of switched-on servers, which in turn minimises the operational expenditure of the operator [147, 216].

5.1.11. *Load Balancing*

Load balancing evenly distributes traffic across network resources to improve SFC performance. 3 studies in the literature aimed to maximise the load balance across network resources.

5.1.12. *Traffic Congestion*

Minimising traffic congestion on network links was used as an optimisation objective in 6 studies. Traffic congestion creates bottlenecks, negatively impacting the performance of SFCs.

5.1.13. *Availability*

Availability is the satisfactory operation of an SFC at a given point in time. 7 studies aimed to maximise availability in dynamic network environments (Section 7.3). The dynamic environments considered included dynamic network topologies [58, 144, 195, 204, 220], where nodes and links could fail, and dynamic SFCRs [68, 213], where SFCRs continued to arrive.

5.1.14. *Other Objectives*

We also found 11 objectives that were used only by 1 study each. Bellavista et al. [221] minimised the worst-case Cut Load Ratio, which partitioned network hosts into subsets and considered the load of the subsets to mitigate poor performance. Ren et al. [222] focused on minimising the virtual topology size by optimising the VNF-CC sub-problem. The authors argued that a smaller virtual topology can be more efficiently embedded on the physical network. Li et al. [100] minimised Service Level Agreement violations, considering performance degradation. Erbati et al. [114] maximised the reliability of SFCs in mobile networks. Zheng et al. [223] maximised the aggregate performance across SFCs. The authors defined an ideal performance as a VNF's performance on a host with no other VNF and an interfered performance as a VNF's performance when it is co-located with other VNFs. A performance drop index was defined to relate the two variables. Xia et al. [224] minimised the number of placement groups of SFCs. A placement group is a modular, self-contained shipping container with servers, networking and storage resources. There can

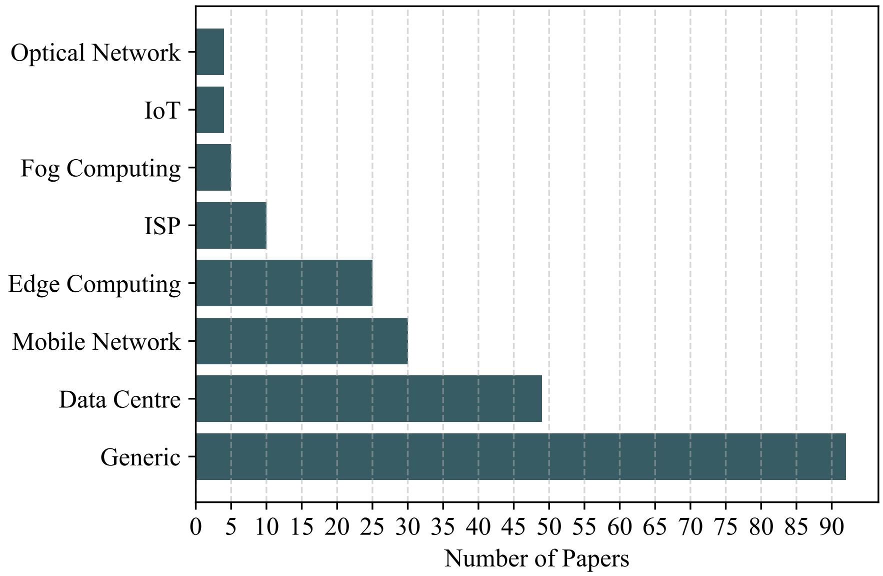

Figure 5: Application domains of the SFC-OCE approaches with more than one study.

be many such placement groups forming the physical network, and the goal here was to minimise the number of placement groups in which the VNFs of an SFC were embedded. Fulber-Garcia et al. [184] aimed to map cache functions to data centres that were geographically distant. Addad et al. [225] minimised the number of VNFs hosting network functions belonging to different network slices in a 5G network. Kuai et al. [19] maximised fairness to prevent one SFC from having significant priority over others. Tang et al. [137] minimised the delay incurred in constructing digital twins of VNFs. Toumi et al. [69] maximised the bandwidth allocated per user in 5G networks.

5.2. Application Domains

SFC is applicable across many application domains such as data centres, 5G, B5G, 6G, edge computing, fog computing, Internet Service Providers (ISPs), the Internet of Things (IoT) and optical networks. SFC-OCE approaches in the literature chose different application domains for implementation and evaluation. We introduce a categorisation that classifies the application domains in the literature into 8 domains, namely, optical networks, IoT, ISP, generic, fog computing, edge computing, data centre, and mobile network. Table 5 classifies the studies according to their application domain.

5.2.1. Optical Network

An optical network uses optical fibre communication for efficient data transfer. 4 studies in the literature used optical networks as their application domain.

Application Domain	**Studies**
Optical Networks	[23, 66, 168, 224]
Internet of Things	[78, 135, 137, 222]
Internet Service Provider	[15, 24, 33, 54, 80, 141, 163, 173, 180, 219]
Generic	[19–22, 34, 39, 40, 44, 45, 55, 57–60, 64, 69, 70, 72, 76, 77, 83, 84, 86–88, 90, 91, 93, 95, 101, 110–123, 126, 129, 130, 132, 136, 138, 139, 146, 150, 152, 153, 156–159, 161, 162, 164, 165, 167, 169, 171, 174–177, 179, 182, 183, 186, 188, 189, 194, 198–202, 205, 206, 208–212, 215, 223, 226]
Fog Computing	[38, 67, 68, 108, 109]
Edge Computing	[31, 36, 37, 56, 61–63, 75, 82, 98–100, 102–106, 124, 142, 148, 149, 168, 187, 197, 213]
Data Centre	[24, 43, 46–54, 65, 73, 74, 82, 85, 124, 125, 127, 128, 131, 140, 143–145, 147, 151, 155, 160, 166, 172, 178, 184, 190–193, 195, 196, 204, 207, 217, 218, 220, 221, 224, 227–229]
Mobile Network	[29–32, 35, 36, 38, 41, 42, 71, 79, 81, 89, 94, 96–98, 133, 134, 154, 168, 170, 181, 185, 203, 213, 214, 216, 225, 230]

Table 5: Classification of studies based on their application domains.

Xia et al. [224] considered SFC embedding on a data-centre network with an optical steering domain serving as the backbone. Riera et al. [23] considered SFC embedding over an optical substrate network. Ruiz et al. [168] performed SFC embedding on a metro optical network that uses Wavelength Division Multiplexing (WDM), equipped with Multi-access Edge Computing (MEC) resources. Khatiri et al. [66] developed an SFC embedding approach on an Elastic Optical Network (EON), which provides better spectrum utilisation than WDM.

5.2.2. Internet of Things

IoT devices are low-cost devices that form a wireless sensor network [78] to capture data about their environment. SFC embedding was applied to IoT in the literature to facilitate communication between devices in IoT networks [135]. 4 studies used IoT as the application domain for their SFC-OCE approach.

5.2.3. Internet Service Provider

10 studies used an ISP as the application domain for their SFC-OCE approach. While Addis et al. [163] considered deploying VNFs in the carrier networks of ISPs, most studies [15, 24, 54, 80, 141, 180, 219] used an ISP topology, such as those generated by RocketFuel (Section 5.3), for evaluating their approaches.

5.2.4. Generic

The majority of the studies, 92 of them, did not consider a specific application domain. Instead, they developed and evaluated their approach for a generic scenario. However, some generic studies evaluated their approach on ISP topologies [76, 77, 165], but since they did not explicitly intend to develop or evaluate their approach on an ISP network, we consider the application domain of such studies as generic.

5.2.5. Fog Computing

Fog computing extends cloud computing from the core to the edge of the network, enabling heavy processing to be offloaded to fog nodes, reducing energy consumption in mobile devices and improving the Quality of Experience of users [38]. 5 studies in the literature used fog computing as the application domain of their approach.

5.2.6. Edge Computing

Edge computing deploys computing capabilities at the edge of the network [148], enabling service providers to satisfy ultra-low latency requirements [142]. This differs from fog computing by placing computation capability only at the edge, in contrast to fog computing, which deploys from the core cloud to the edge. 25 studies used edge computing as the application domain for their approach.

5.2.7. Data Centre

Data Centre is the most popular application domain in the literature for SFC-OCE approaches. A data centre consists of computing and networking infrastructure, on top of which SFCs can be embedded. 49 studies used data centres as their application domain for designing or evaluating their approaches.

5.2.8. Mobile Network

Mobile network was the second most popular application domain for the SFC-OCE approaches in the literature, with 30 studies using it as their application domain. Within this domain, 5G, B5G, and 6G are the most popular use cases [29–31, 35, 36, 79, 81, 89, 89, 94, 96–98, 133, 134, 168, 170, 185, 203, 213, 216, 225], an expected observation given SFC's use case in network slicing. 4 studies [38, 107, 230] considered a Radio Access Network (RAN) as their application domain for their approach. RAN is a communication system that connects user equipment to the mobile core via radio links. 4 studies [41, 42, 154, 181] used the mobile core network as the application domain for their approach. Apart from these, Bagaa et al. [214] used a carrier cloud as their application domain. A carrier cloud virtualises network functions and networks over a federated cloud [214].

5.2.9. Other Domains

Hentati et al. [92] used remote robotic surgery as the application domain for their SFC-OCE approach. The authors developed an SFC-OCE approach for a 5G-enabled tactile remote robotic system that catered to the different Quality of Service requirements of different traffic data [92]. Riggio et al. [107] developed their approach for enterprise WLANs.

5.3. Network Topologies

The physical networks on which SFCs were embedded used multiple different topologies in the literature, as shown in Fig. 6. Fat tree topology was the most popular topology in the literature, with 20 studies [24, 46, 49, 65, 69, 70, 73, 85, 89, 113, 140, 144, 147, 150, 152, 186, 190, 191, 196, 230] using it for the substrate network. Fat tree topologies are hierarchical topologies with three layers and are commonly used in data centre networks [231]. NSFNET was the second most popular topology, which was used by 11 studies [37, 47, 51, 53, 63, 66, 77, 79, 142, 170, 174]. NSFNET is a high-speed backbone network created in the United States to connect university research networks [232]. 10 studies [37, 82, 84, 88, 135, 149, 155, 156, 175, 186] used random network topologies, where the connections between nodes were determined by a probability. 9 studies [33, 45, 74, 94, 108, 139, 196, 206, 219] used GEANT, which is a pan-European network, interconnecting research and educational networks across Europe [233]. BTEurope, British Telecom's European network, was used by 8 studies [44, 55, 76, 80, 94, 193, 194, 203]. 8 studies [125, 127, 139, 164, 192, 195, 209, 219] used the Abilene network, which was a high-speed backbone network in the United States until it was retired in 2007. BCube [234], a network topology for modular

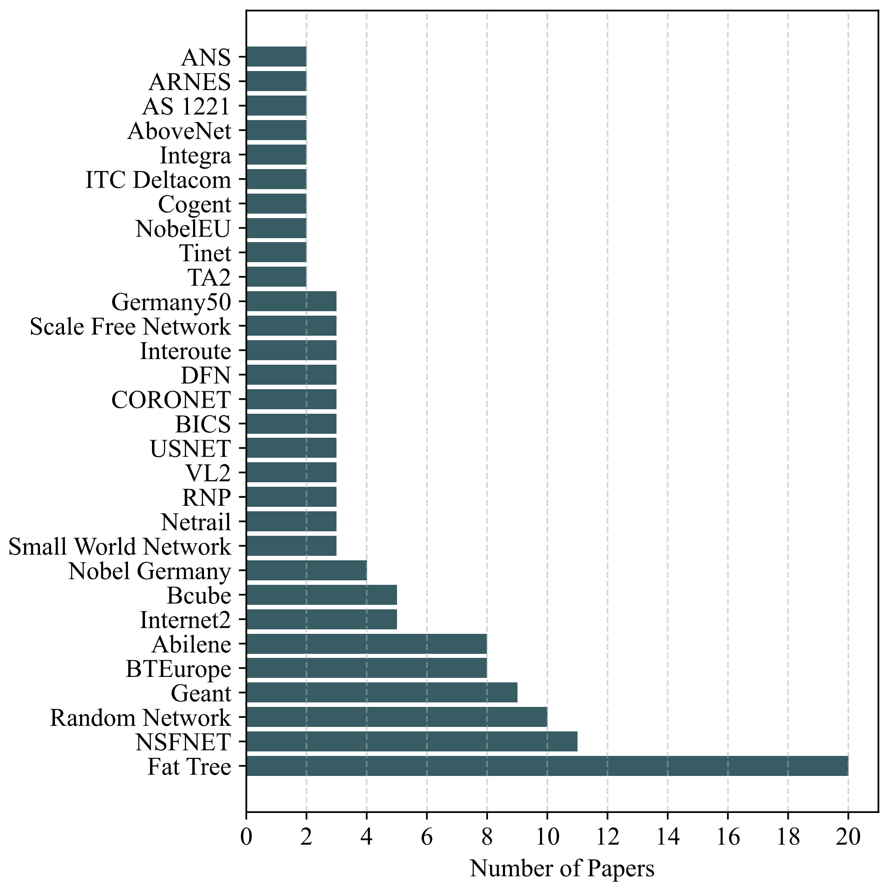

Figure 6: Network topologies that have been used by at least one study in the literature.

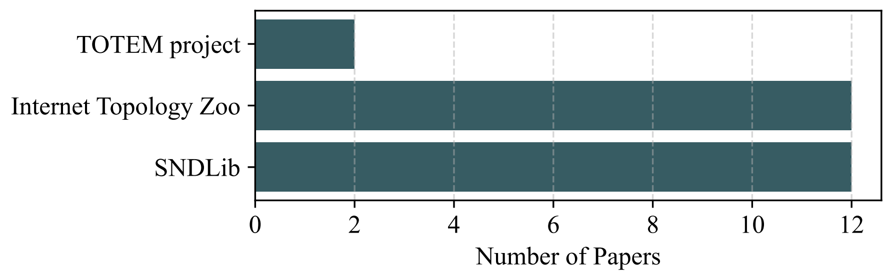

Figure 7: The topology libraries used in the literature.

data centres, was used by 5 studies [85, 144, 147, 190, 191]. Internet2, which is an upgraded version of Abilene [235], was also used by 5 studies [54, 149, 164, 206, 207].

The other network topologies used include Nobel Germany [90, 95, 137, 138], CORONET [43, 153, 227], VL2 [147, 190, 191], Small World Network [84, 88, 135], Scale Free Network [84, 88, 135], RNP [100, 125, 204], Netrail [193, 194, 203], Interoute [44, 55, 80], DFN [72, 82, 101], and BICS [100, 192, 195]. Additionally, USNET [79, 79, 153, 207], TINET [89, 101], TA2 [50, 101], Integra [192, 195], Germany50 [41, 125, 126], Cogent [52, 191], Abovenet [192, 195], ANS [192, 195], NobelEU [42, 206], and AS 1221 [141, 180] , ARNES [82, 101], ITC Deltacom [82, 95] were also used.

5.3.1. Topology Libraries

Finding network topologies for networking experiments is often challenging. Therefore, we now present the network topology libraries that we discovered during our literature analysis. Fig. 7 shows the topology libraries used in the literature. Internet Topology Zoo was used by 12 studies [24, 44, 55, 72, 80, 82, 94, 100, 101, 104, 138, 219]. Topologies like BTEurope, ARNES, and Deltacom were obtained from this library. SNDlib was also used by 12 studies [41, 42, 45, 74, 101, 108, 125, 127, 138, 160, 178, 209]. Topologies like GEANT and Abilene were obtained from SNDlib. Additionally, topologies were also taken from the TOTEM project by 2 studies [20, 57].

5.3.2. Topology Models

Three topology models were used in the literature to generate network topologies, as shown in Fig. 8. The Barabasi-Albert model, which is used for generating scale-free topologies, was used by 8 studies [15, 33, 56, 156, 157, 175, 179, 186]. The Waxman model, which is used for generating random graphs, was also used by 8 studies [34, 158–160, 169, 177, 202, 212]. The Erdos-Renyi model, which is also used for generating random graphs, was used by 2 studies [56, 126].

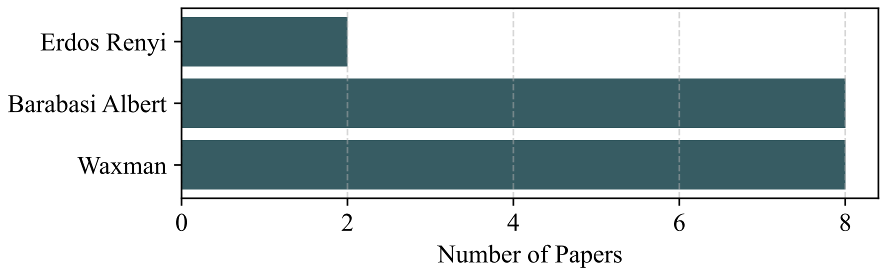

Figure 8: The topology models used in the literature.

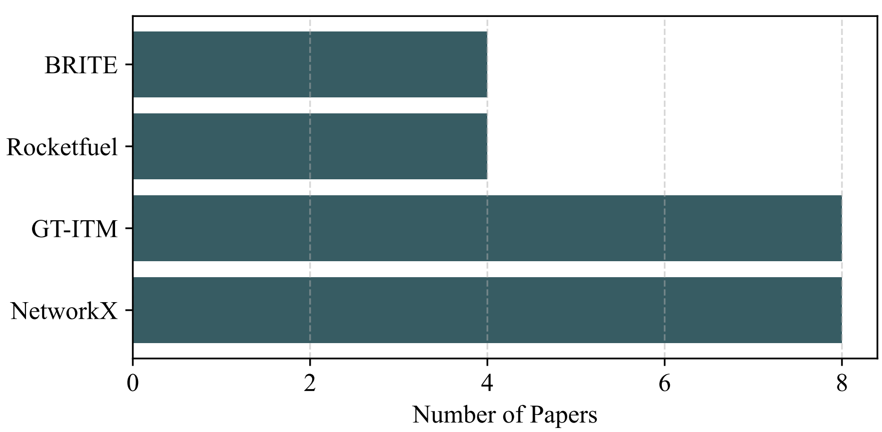

Figure 9: The topology generators used in the literature.

5.3.3. Topology Generators

Four topology generators were used in the literature to generate network topologies, as shown in Fig. 9. NetworkX, which is a Python-based graph generator, was used by 8 studies [43, 56, 58, 64, 70, 80, 113, 135]. GT-ITM, which contains routines to generate graphs, was also used by 8 studies [39, 55, 81, 89, 106, 110, 182, 211]. BRITE, a Java and C++-based generator, was used by 4 studies [15, 71, 158, 160]. Rocketfuel [236], which generates ISP network topologies, was used by 4 studies [54, 141, 165, 180].

6. Classification of Approaches by SFC-OCE Sub-problems

The SFC-OCE problem consists of three sub-problems, viz. VNF-CC, VNF-EM, and VL-EM (Section 4). In this section, we classify the SFC-OCE approaches by the sub-problem, or the set of sub-problems they optimised. A summary of this classification is given in Table 6. Most approaches considered only one or two of the sub-problems, while only a few optimised all three sub-problems. To produce an optimal solution, an approach has to optimise all three sub-problems simultaneously. This is an $\mathcal{NP}$-hard optimisation problem.

7 studies optimised only the VNF-CC sub-problem. 52 studies optimised only the VNF-EM sub-problem. 124 studies optimised both the VNF-EM and VL-EM sub-problems, out of which 35 studies proposed a sequential approach, while 89 studies proposed a simultaneous approach. Among all SFC-OCE approaches, optimising VNF-EM and VL-EM simultaneously was the most popular. Approaches that optimised both VNF-EM and VL-EM sequentially either solved VL-EM first, followed by VNF-EM, or vice versa. Approaches that optimised VL-EM first generally found the shortest path between the source and destination nodes in the network and then embedded VNFs on the servers along the shortest path [35, 53, 79, 97, 160, 183, 208]. Approaches that optimised VNF-EM first, embedded VNFs on the optimal servers and then used a shortest-path algorithm, such as Dijkstra, to optimise VL-EM [38, 64, 71, 74, 101, 110].

Only 18 out of the 209 studies optimised VNF-CC, VNF-EM, and VL-EM, out of which 8 studies [79–81, 127, 170, 172, 182, 215] optimised sequentially, while 10 studies [20, 84, 91, 94, 169, 171, 174, 175, 179, 199] optimised simultaneously. Fig. 10 shows a Venn diagram with the number of studies by the sub-problems they optimised.

When considering the NFV-RA problem along with the SFC-OCE problem, we found 8 studies [19, 21–23, 122, 124, 128, 131] that optimised only the VNF-SCH sub-problem of NFV-RA. 2 studies [119, 132] optimised VNF-EM and VNF-SCH sequentially, while 5 studies [111, 116, 118, 121, 129] optimised them simultaneously. Only 1 study [137] optimised VNF-SCH, VNF-EM and VL-EM sequentially, while 2 studies [92, 201] optimised them simultaneously. 2 studies [20, 84] optimised VNF-CC, VNF-EM, VL-EM and VNF-SCH simultaneously. 1 study [123] optimised VNF-CC and VNF-SCH simultaneously. In total, we found 21 studies that optimised VNF-SCH.

VNF-CC	VNF-EM	VL-EM	Simultaneous	Studies	No. of Studies
✓	X	X	N/A	[83, 93, 123, 167, 176, 222, 226]	7
X	✓	X	N/A	[37, 45, 46, 49, 52, 58, 61, 63, 67, 68, 70, 75, 82, 86–88, 96, 99, 102, 105, 108, 109, 111, 113–121, 129, 130, 132, 142, 146, 148, 173, 181, 184, 197, 203–205, 210, 214, 216, 221, 223, 224, 228]	52
X	✓	✓	✓	[15, 24, 29–31, 33, 34, 41–44, 47, 48, 50, 51, 54–57, 62, 65, 66, 69, 72, 73, 76–78, 89, 90, 92, 95, 100, 103, 104, 106, 107, 125, 126, 133–136, 138, 139, 143–145, 150–157, 159, 161–164, 166, 168, 178, 185–195, 198, 200–202, 212, 213, 217–220, 225, 227, 229, 230]	89
X	✓	✓	X	[32, 35, 36, 38–40, 53, 59, 60, 64, 71, 74, 85, 97, 98, 101, 110, 112, 137, 140, 141, 147, 149, 158, 160, 165, 177, 180, 183, 196, 206–209, 211]	35
✓	✓	✓	✓	[20, 84, 91, 94, 169, 171, 174, 175, 179, 199]	10
✓	✓	✓	X	[79–81, 127, 170, 172, 182, 215]	8

Table 6: A summary of the SFC-OCE approaches classified by the SFC-OCE sub-problems.

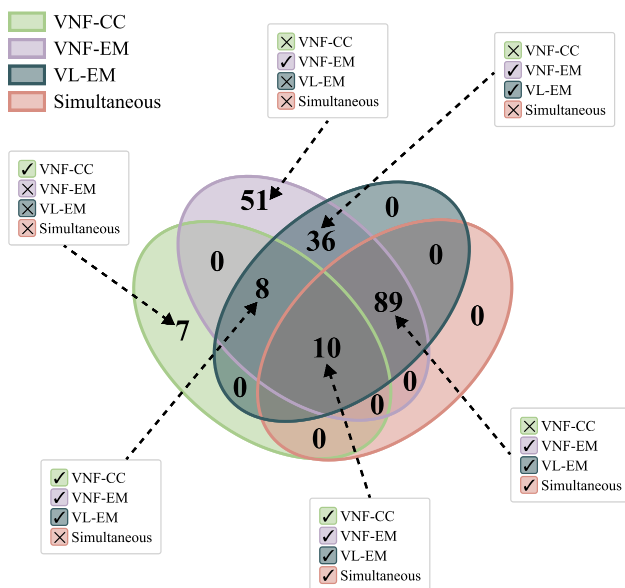

Figure 10: A Venn diagram showing the number of studies by the sub-problems they optimised. Most of the approaches (89) optimised VNF-EM and VL-EM simultaneously. Only 10 studies optimised all three sub-problems simultaneously, shown by the 10 in the centre of all ovals.

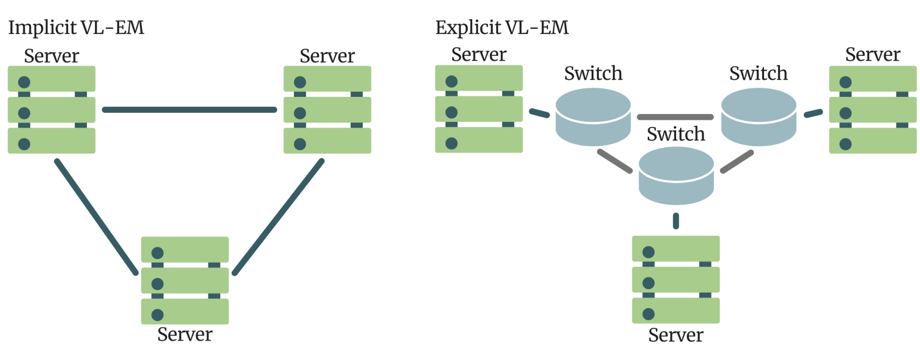

Figure 11: The network on the left is an abstract topology, where direct physical links are assumed between servers, and the one on the right is a full topology, where servers are connected via switches. Implicit VL-EM approaches considered an abstract topology, whereas explicit approaches considered a full topology.

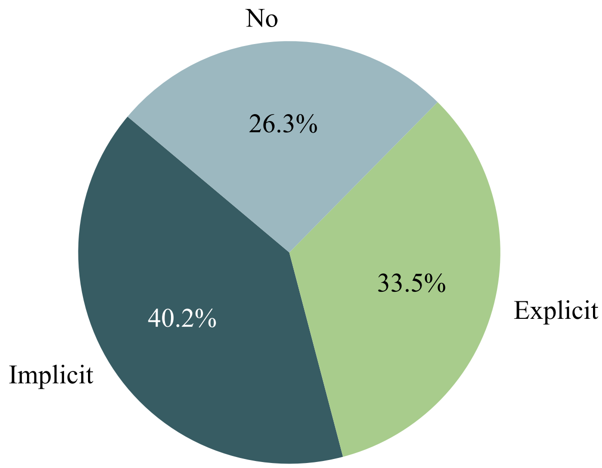

Figure 12: A pie chart showing how approaches that optimised the VNF-EM problem tackled the VL-EM problem. 40.2% of the studies (78) optimised VL-EM implicitly. 33% (65) optimised VL-EM explicitly. 26.8% (52) did not optimise VL-EM.

6.1. Explicit vs. Implicit VL-EM

We observed that the majority of the approaches that optimised VL-EM considered an abstract network topology. An abstract topology assumes that there exist direct physical links between the servers in the topology, as shown in Fig. 11. This greatly reduces the complexity of the VL-EM sub-problem, as the approach does not need to find the optimal path between servers hosting two successive VNFs in an SFC via network switches. While this approach can be useful for initial exploration of the problem, it produces an incomplete VL embedding, as it does not specify where the VLs should be embedded on the physical links between switches. Besides, these approaches are not guaranteed to produce an optimal solution, as the performance of the network switches and the links between them are not considered. Consequently, we consider VL-EM approaches based on abstract networks to be *implicit*.

An *explicit* VL-EM approach considers a complete topology where the servers are connected via network switches. Explicit approaches produce complete and optimal solutions to the VL-EM sub-problem. We also consider approaches that use a shortest-path algorithm, such as Dijkstra, to find the shortest path between two servers on an abstract topology, to be explicit as well, as the shortest-path algorithm can find a path between servers via switches when applied to a full topology.

Some approaches that considered a full topology failed to detail their VL-EM approach [127, 220]. We consider such approaches to be implicit. Fig. 12 shows a pie chart showing how the approaches that optimised VNF-EM approached the VL-EM sub-problem. 78 studies optimised the VL-EM sub-problem implicitly. Explicit optimisation was carried out by 65 studies. However, 51 of the 65 explicit approaches used a shortest-path algorithm to find a path between servers

VL-EM Approach	**Studies**
Implicit	[15, 20, 24, 29–33, 35, 36, 39–42, 44, 50, 51, 53, 59, 60, 69, 76, 78, 79, 90, 92, 94, 95, 97, 100, 104, 106, 126, 127, 136, 138, 139, 145, 149–157, 160, 162, 164, 166, 169–171, 177, 178, 182, 183, 185–188, 192–196, 199, 202, 207, 208, 213, 215, 218–220, 225, 227]
Explicit	[34, 38, 43, 47, 48, 54–57, 62, 64–66, 71–74, 77, 80–82, 84, 85, 89, 91, 98, 101, 103, 107, 110, 112, 125, 133–135, 137, 140, 141, 143, 144, 147, 158, 159, 161, 163, 165, 168, 172, 174, 175, 179, 180, 189–191, 198, 200, 201, 206, 209, 211, 212, 217, 229, 230]
Explicit using a shortest-path algorithm	[34, 38, 48, 54–56, 64, 65, 71, 72, 74, 77, 80–82, 84, 85, 89, 91, 98, 101, 103, 107, 110, 125, 133–135, 137, 140, 141, 147, 158, 165, 168, 172, 174, 175, 179, 180, 189–191, 200, 201, 206, 209, 211, 212, 229, 230]

Table 7: Classification of studies that optimised VL-EM according to the type of their approach.

hosting two successive VNFs. Using only the shortest path to link servers creates asymmetric traffic, leading to congestion on links on the shortest path, while other links remain underutilised [194]. Among the 10 approaches that optimised all three sub-problems simultaneously, only 5 approaches [84, 91, 174, 175, 179] optimised VL-EM explicitly. However, all 5 approaches used a shortest-path algorithm for VL-EM. Table 7 shows studies that optimised VL-EM classified according to their approach type.

7. Algorithms Used

In this section, we analyse the SFC-OCE approaches in terms of the algorithms they used. We initially give an overview, and then provide a discussion in the subsections.

We found 15 different types of algorithms in the literature which were used by more than one study. To simplify our analysis, we grouped all heuristic algorithms and custom exact algorithms into two respective groups. Fig. 13 shows a summary of the algorithms found in the literature and the number of studies that used them. Table 8 classifies the approaches according to the algorithms used. Some approaches used a hybrid of two algorithms (Section 7.1). We classified such approaches under both algorithms.

23 studies used a custom exact algorithm to optimise the SFC-OCE problem. Linear Programming, which is an exact algorithm, was used by 19 studies. Dynamic programming, which breaks a complex problem down to sub-problems and produces an optimal solution, was used by 4 studies, out of which 2 used

Algorithm	Studies
Custom Exact	[52, 72, 82, 84, 94, 109, 129, 132, 142, 145, 151, 152, 154, 171, 174, 181, 187, 189, 191, 207, 222, 227, 228]
Linear Programming	[37, 41, 42, 50, 51, 74, 90, 97, 134, 159, 161, 164, 166, 167, 178, 182, 185, 199, 217]
Custom Dynamic Programming	[130, 148]
Viterbi	[54, 88]
Hungarian	[106, 149]
Heuristic	[15, 29–32, 35, 36, 38–40, 43, 44, 46–49, 55, 59–61, 65, 66, 68, 79–81, 83, 85, 92, 93, 95, 98, 99, 107, 114, 126, 127, 134, 141, 144, 150, 153, 156, 160, 162, 163, 165, 169, 172, 175, 177, 179, 180, 182, 183, 186, 190, 196–198, 206, 209–211, 213–215, 218, 220, 221, 223–225, 229, 230]
Greedy Algorithm	[24, 33, 71, 75, 121, 130, 137, 140, 188, 214]
Column Generation	[77, 123]
Reinforcement Learning	[19, 20, 34, 57, 58, 62, 64, 67, 70, 73, 76, 78, 91, 100, 102, 103, 105, 108, 113, 115, 125, 135, 136, 139, 143, 157, 182, 192–195, 200–205, 212]
Machine Learning	[20, 89, 96, 98, 102, 103, 110, 115, 117]
Game Theory	[53, 104, 120, 122, 131, 133, 219]
Genetic Algorithm	[21, 45, 50, 69, 74, 86, 111, 116, 118, 119, 128, 131, 140, 147, 168, 170, 173, 184, 208]
Grey Wolf Optimisation	[101, 112]
Tabu Search	[56, 121, 137, 158, 176]

Table 8: Classification of approaches based on the algorithms used.

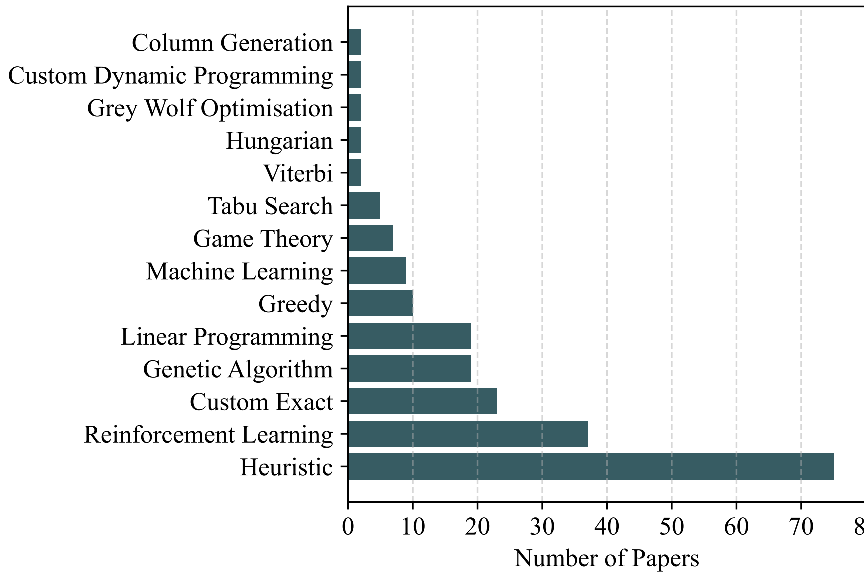

Figure 13: The algorithms used in SFC-OCE approaches and the number of studies that used them. Only algorithms that were used by more than one study were considered.

custom dynamic programming-based algorithms, while the other 2 used Viterbi. 2 studies used the Hungarian algorithm, which is another exact algorithm. Exact algorithms are guaranteed to produce an optimal solution. However, when it comes to $\mathcal{NP}$-hard optimisation problems, they take prohibitively longer time to produce an optimal solution when the complexity of the problem increases. As a result, exact algorithms do not scale well to complex SFC-OCE problems, requiring heuristic or meta-heuristic algorithms. Some studies proposed an exact algorithm to establish its limited scalability and then proposed a heuristic algorithm to overcome it [29, 80, 114, 144, 156, 160, 162, 165, 180, 218].

Heuristic algorithms were used by 75 studies, making them the most popular algorithm in the literature. Heuristic algorithms are used for complex $\mathcal{NP}$-hard optimisation problems to produce a near-optimal solution within an acceptable amount of time. Greedy algorithms, a type of heuristic algorithm which makes the locally optimal decision at every step, were used by 10 studies. Column Generation algorithms that use heuristics were used by 2 studies.

Reinforcement Learning (RL) was the second most popular algorithm, with 38 studies using it. RL is a machine learning algorithm where an agent learns to interact with its environment via rewards and penalties accumulated through an iterative trial-and-error process. We observed multiple variations of RL algorithms used in the literature. 4 studies [73, 103, 115, 193] used Q-Learning based RLs. The Advantage Actor-Critic-based RLs were used by 1 study [204], while the Asynchronous Advantage Actor-Critic-based (A3C) RLs were used by

8 studies [34, 62, 67, 100, 108, 125, 143, 212]. 4 studies [19, 64, 200, 204] used the Proximal Policy Optimisation-based RLs. The Deep Q-Network (DQN) was used by 6 studies [70, 76, 78, 105, 135, 157]. Double DQN was used by 6 studies [57, 136, 192, 194, 195, 203]. Duelling DQN [139], Duelling Double DQN [201], and Branching Duelling Q-Network [91] were used by 1 study each. Additionally, Machine Learning (ML) algorithms were used by 9 studies, driven by the rapid advancement in deep learning. Pandey et al. [103, 115] used Recurrent Neural Networks (RNNs) along with RL, while Zou et al. [102] used Long Short-Term Memory (LSTM) networks with DQN. Neural Networks were used by Subramanya et al. [98], Li et al. [20] and Xu et al. [237].

Game theory, which uses mathematical models of strategic interaction between multiple players, was used by 7 studies. Meta-heuristic algorithms like Genetic Algorithms (GAs), Grey Wolf Optimisation (GWO) and Tabu Search algorithms were also used in the literature. GAs that imitate evolution in nature by performing crossovers and mutations were used by 19 studies. 2 studies used GWO, which is based on the hierarchy and hunting behaviour among grey wolves. Tabu Search was used by 5 studies.

Other algorithms used include Swarm Intelligence [155], Successive Convex Approximation [198], Stable Matching [216], Non-Linear Programming [49], Monte Carlo Tree Search [182], Mixed-Integer Quadratic Constrained Programming [127], Meteor Shower Optimisation [146], Markov Decision Process (MDP) [89], Finite Automaton [226], Discrete Spider Monkey Optimisation [138], blockchain [62], Bee Colony Optimisation [74], and Simulated Annealing [87].

7.1. Hybrid Algorithms

8 approaches used a hybrid of two algorithms to optimise SFC-OCE. Pandey et al. used an RNN to predict the resource demands in advance and then used Q-learning-based RL to optimise SFC-OCE [103, 115]. Subramanya et al. used a Neural Network (NN) to predict the required number of VNF instances based on traffic demand and used ILP and a heuristic algorithm to optimise SFC-OCE [98]. Xu et al. used NN and MDP to optimise SFC-OCE [237]. Li et al. used NN and Deep RL to optimise SFC-OCE [20]. Zou et al. used LSTM to predict network nodes that may be attacked and DQN to optimise SFC-OCE in a risk-aware manner [102]. Guo et al. used blockchain to establish trust between smart devices and service providers in distributed networks, and A3C-based RL to optimise SFC-OCE [62]. Yuan et al. used GA to optimise VNF-SCH, while using game theory for bandwidth allocation [131].

7.2. Scalability of Algorithms

The SFC-OCE problem is an $\mathcal{NP}$-hard optimisation problem that becomes extremely complex in real networking environments. In a real networking environment, the network topology can have a very high number of servers and switches, and the number of SFCRs a service provider receives can be very high. Exact algorithms will not scale well to such scenarios, requiring more scalable algorithms. On that account, in this section, we analyse the approaches in the

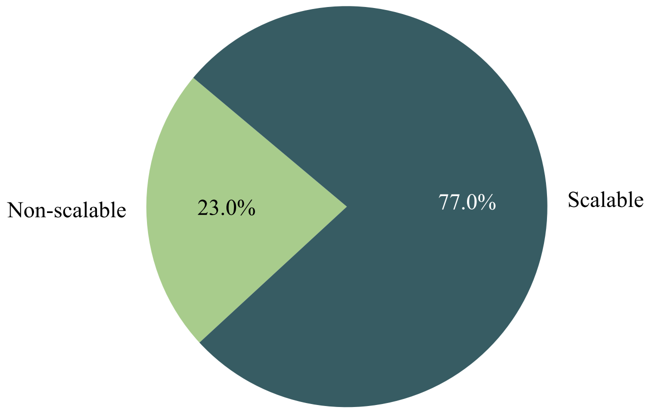

Figure 14: Percentage of scalable vs. non-scalable approaches in the literature. 161 (77%) approaches are scalable, while 48 (23%) are not scalable.

Scalability	**Studies**
Scalable	[15, 19–21, 24, 29–36, 38–40, 43–50, 53, 55–62, 64–71, 73–81, 83, 85–87, 89, 91–93, 95, 96, 98–105, 107, 108, 110–123, 125, 126, 128, 130, 131, 133–141, 143, 144, 146, 147, 150, 153, 155–158, 160, 162, 163, 165, 168–170, 172, 173, 175–177, 179, 180, 182–184, 186, 188, 190, 192–198, 200–206, 208–216, 218–221, 223–225, 228–230]
Non-scalable	[22, 23, 37, 41, 42, 51, 52, 54, 63, 72, 82, 84, 88, 90, 94, 97, 106, 109, 124, 127, 129, 132, 142, 145, 148, 149, 151, 152, 154, 159, 161, 164, 166, 167, 171, 174, 178, 181, 185, 187, 189, 191, 199, 207, 217, 222, 226, 227]

Table 9: Studies by their scalability.

literature in terms of their scalability. We consider all approaches that used exact algorithms, which include the Hungarian algorithm, custom exact algorithms, linear programming, custom dynamic programming, and Viterbi, to be lacking in scalability. We consider all approaches that used meta-heuristic algorithms, such as GAs, GWO, and Tabu Search, heuristic algorithms, including Column Generation and greedy algorithms, game theory-based algorithms, machine learning and RLs to be scalable algorithms. Consequently, as shown in Fig. 14 and Table 9, 161 studies proposed a scalable approach, whereas the approaches of 48 studies were not scalable.

7.3. Static vs. Dynamic Algorithms

Algorithms used in SFC-OCE approaches can be classified into static and dynamic based on the nature of the network environment they considered. Dy-

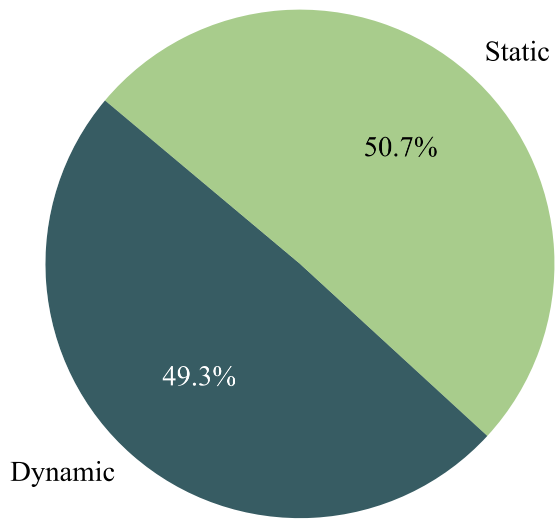

Figure 15: 50.7% (106) approaches used a dynamic algorithm, whereas 49.3% (103) approaches used a static approach.

namic algorithms consider a dynamic network environment, where the network topology, the amount of traffic received, or the number of SFCRs received could vary with time. Static algorithms consider a snapshot of a network where the topology and the traffic remain constant, while only a static set of SFCRs has to be embedded. Since real network environments are invariably dynamic, dynamic algorithms are more likely to be useful in real networks. We observed that 106 studies were static, while 103 were dynamic, as shown in Fig. 15. We classify dynamic approaches into dynamic topology, dynamic SFCRs, and dynamic traffic based on the dynamic factors in the network environment.

7.3.1. Dynamic SFCRs

Most SFC-OCE approaches considered only a static set of SFCRs. However, in a real network, SFCRs can continue to arrive, and the approach should be able to optimally embed the arriving SFCRs. In the literature, the arrival of SFCRs generally followed a Poisson distribution [89, 94, 133, 137, 141, 175, 177, 182, 198, 202, 203, 216]. However, Chen et al. used the power law distribution [183]. 73 studies in the literature considered dynamic SFCRs.

7.3.2. Dynamic Topology

A network topology can be dynamic when the network topology changes over time due to server and link failures. Some of the approaches that considered a dynamic topology optimised the availability of SFCs so that SFC embeddings would adapt to topological changes [144, 195, 204, 220]. 16 studies in the literature considered a dynamic topology.

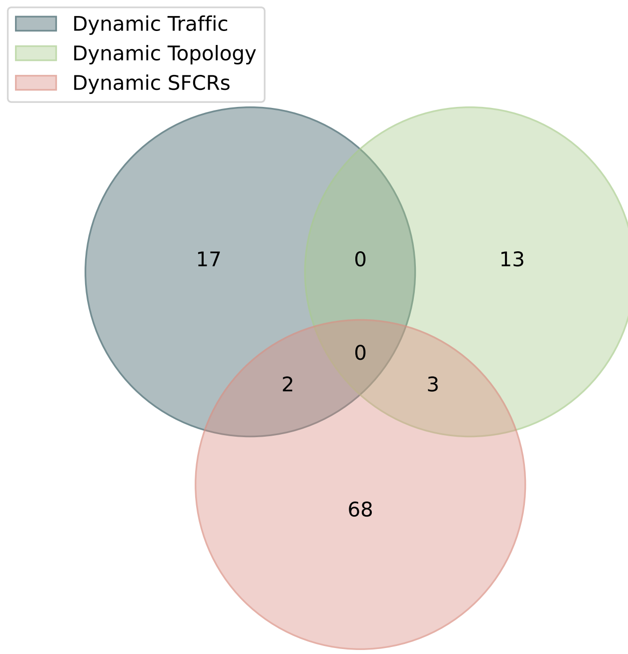

Figure 16: Dynamic approaches by the dynamic factor they considered.

7.3.3. Dynamic Traffic

The traffic requests to the SFCs are dynamic in a real network environment. In the literature, dynamic traffic was generated randomly [178], based on real usage data [181], based on a Gaussian distribution [135] or based on a Poisson distribution [92]. 19 studies considered dynamic traffic in their approach.

Table 10 classifies the algorithms in the literature according to the nature of the environment they considered. Fig 16 shows the number of studies by the dynamic factor they considered.

8. Evaluation Mechanisms

To evaluate possible solutions to the SFC-OCE problem, different evaluation mechanisms were used. We classify the evaluation mechanisms used in the literature into numerical evaluation, simulation, emulation, Production-Grade Virtualisation Tools (PGVTs), and testbeds. Fig. 17 provides a summary of the percentage of studies that used the different types of evaluation mechanisms.

8.1. Numerical Evaluation

Numerical evaluation was the most popular evaluation mechanism, with 164 studies using it. To evaluate embeddings, numerical evaluation uses mathematical models of computer networks, traffic patterns and SFCs, providing fast

Dynamic Nature	**Studies**
Static	[15, 19, 21–24, 30–32, 35, 37, 39, 41–45, 48–54, 56, 61, 69–71, 73–75, 79, 82–84, 87, 88, 91, 93, 95, 97, 99, 102, 105–107, 109, 111–113, 117–119, 124, 127, 130, 131, 134, 140, 143, 150, 152–157, 161–171, 173, 174, 176, 180, 184–188, 193, 196, 197, 200, 201, 205–207, 214, 215, 217, 219, 223–227, 230]
Dynamic Traffic	[33, 36, 46, 77, 92, 96, 98, 103, 115, 120, 135, 141, 147, 178, 181, 208, 210, 221, 228]
Dynamic SFCRs	[20, 29, 34, 38, 40, 46, 47, 55, 57, 59, 60, 62–68, 72, 76, 78, 80, 81, 85, 89, 90, 94, 100, 101, 104, 108, 110, 116, 121–123, 125, 129, 132, 133, 136–139, 141, 142, 145, 149, 158–160, 172, 175, 177, 182, 183, 189–192, 194, 198, 199, 202, 203, 209, 211–213, 216, 218, 222, 229]
Dynamic Topology	[57, 58, 64, 86, 114, 126, 128, 144, 146, 148, 151, 179, 195, 204, 212, 220]

Table 10: Studies classified according to the nature of the network environment considered.

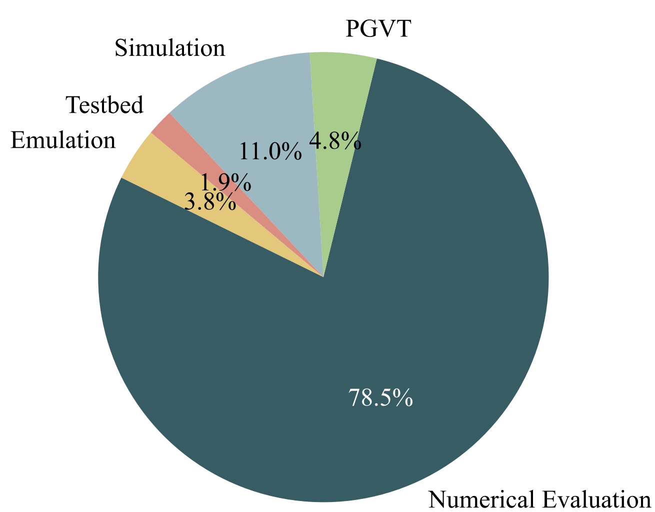

Figure 17: A summary of the types of evaluation mechanisms used in the literature.

evaluations at the expense of accuracy and fidelity as modern networks are too complex to be accurately modelled [238].

8.2. Simulation

Simulation was used by 23 studies. Simulators create a software model of the network, enabling faster, more scalable evaluation of SFC-OCE approaches compared to emulators and testbeds [239]. However, as is the case with numerical evaluation, software models cannot capture the complexity or the dynamic nature of networks. As a result, despite their speed, simulators offer inaccurate, low-fidelity evaluation. ALEVIN, a Java-based Virtual Network Embedding simulator, was the most popular simulator in the literature, used by 13 studies [106, 145, 148–153, 169, 175, 179, 186, 187]. CloudSim, which simulates cloud infrastructure, was used by 3 studies [65, 144, 173]. OMNet++, a C++-based network simulation framework, was used by 2 studies [168, 208]. A simulator called Very Lightweight Service Platform was used by Clayman et al. [228]. "Simulation Platform for Scotfield Cao" based on ALEVIN was used by Cao et al. [177]. Additionally, a Java-based discrete event simulator [128], and two custom simulators [83, 154] were also used.

8.3. Testbeds

Testbeds were used by 4 studies. Testbeds are physical networks meant for networking experiments. Since they are real networks, they offer the most accurate and high-fidelity evaluation among all types of mechanisms [239]. However, the evaluations are slower, as they are real experiments, while offering poor scalability, as scaling physical networks is both expensive and time-consuming [239]. The 5GinFIRE testbed was used by Bunyakitanon et al. [96]. CREATE-NET was used by Riggio et al. [32], and a Raspberry Pi deployment platform was used by Almurshed et al. [68].

8.4. Emulation

Emulation was used by 8 studies. Emulators duplicate networks using real networking software and virtualisation tools such as Open vSwitch and Docker containers [140]. As they duplicate networks instead of modelling them, they offer better accuracy and higher fidelity than simulators, even though they are slower and less scalable [239]. However, they are faster and more scalable than testbeds, while being less accurate [239]. In other words, emulators offer a trade-off between simulators and testbeds. Mininet, which offers real Open vSwitch, was the most popular emulator in the literature, with 7 studies using it [74, 102, 146, 147, 155, 188, 220]. OpenRASE, which uses Mininet to emulate the physical network and Docker containers to emulate servers, is an emulator meant specifically for SFC-OCE evaluation and was used by 1 study [140].

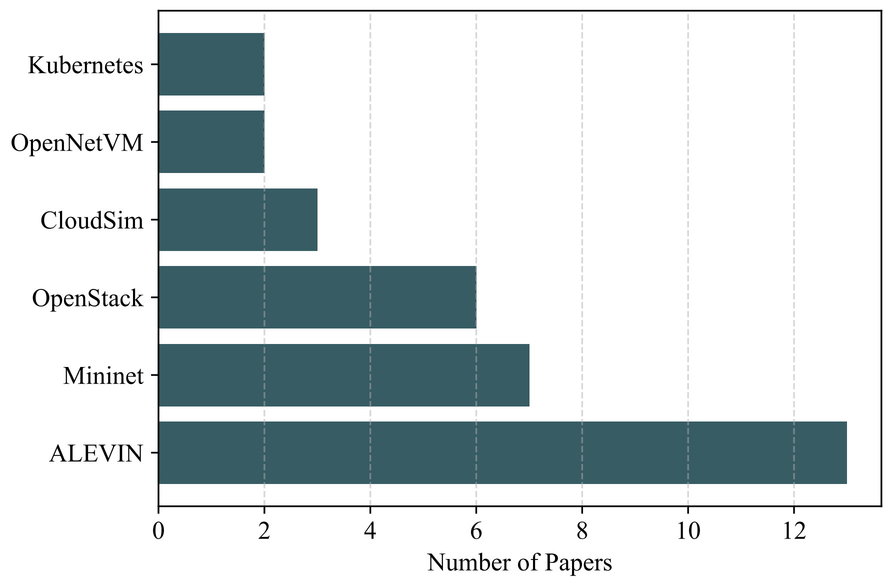

Figure 18: The number of studies using the different evaluation tools found in the literature. Only evaluation tools that are used by more than 1 study are shown.

8.5. Production Grade Validation Tools

PGVTs were used by 10 studies. PGVTs are virtualisation tools, such as Kubernetes, that are used in production environments. They are comparable to emulators as they offer higher fidelity than simulators while being more scalable and faster than testbeds. OpenStack, a cloud computing platform, was used by 6 studies [56, 103, 105, 115, 133, 221]. Kubernetes, which is a container orchestration platform, was used by 2 studies [58, 109]. OpenNetVM, which is an NFV platform, was used by 2 studies [210, 223]. Fig. 18 shows a summary of the evaluation tools used by more than 1 study in the literature. Table 11 classifies the studies by the type of evaluation mechanism they used.

9. Emerging Trends

We observed several emerging trends in the literature, which we discuss in this section.

9.1. VNF Decomposition

A VNF can be decomposed into multiple components, as shown in Fig. 19, and embedded on multiple servers, offering several benefits, such as efficient maintenance and scaling [194]. While this can help take better advantage of the available resources, this adds to the complexity of the SFC-OCE problem, as every component of a decomposed VNF has to be independently embedded,

Evaluation Type	Studies
Numerical	[15, 19–24, 29–31, 33–55, 57, 59–64, 66, 67, 69–73, 75–82, 84–86, 88–95, 97–101, 104, 107, 108, 110–114, 116–127, 129–132, 134–139, 141–143, 156–167, 170–172, 174, 176, 178, 180–185, 189–207, 209, 211–219, 222, 224–227, 229, 230]
Simulation	[65, 83, 106, 128, 144, 145, 148–154, 168, 169, 173, 175, 177, 179, 186, 187, 208, 228]
Testbeds	[32, 68, 87, 96]
Emulation	[74, 102, 140, 146, 147, 155, 188, 220]
PGVTs	[56, 58, 103, 105, 109, 115, 133, 210, 221, 223]

Table 11: Studies classified according to their evaluation mechanisms.

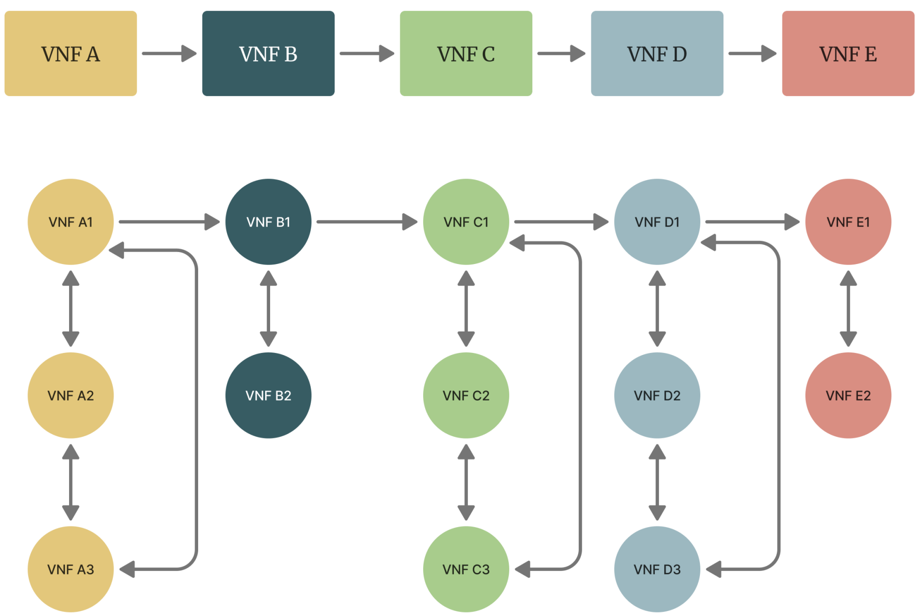

Figure 19: A typical SFC with monolithic VNFs is shown at the top, while an SFC with decomposed VNFs is shown at the bottom.

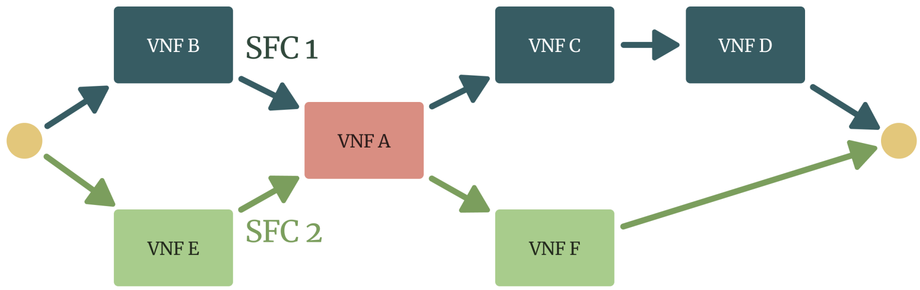

Figure 20: VNF Sharing: Two SFCs, SFC1 and SFC2, share VNF A.

increasing the search space. The SFC-OCE problem with VNF decomposition was considered by 9 studies [44, 80, 83, 90, 93, 135, 194, 198, 203].

9.2. VNF Sharing

VNF sharing involves sharing an instance of a VNF among several SFCs, as shown in Fig. 20. The traffic load across VNFs in a network differs, which means some VNFs may remain idle or have a low demand. In such cases, sharing the idle or low-demand VNF with other SFCs offers several benefits, such as reduced licensing cost [39] and optimal resource utilisation [37]. This concept was used by Yue et al. [150], Wang et al. [240], Mohamad et al. [37, 63, 142], Guo et al. [39], and Zhang et al. [67]. Doan et al. [56] proposed the use of sub-chain sharing. A sub-chain consists of a sequence of VNFs in an SFC, and, in addition to the benefits offered by VNF sharing, it reduces SFC configuration cost by performing route configuration only once.

9.3. Concurrent VNF Execution

Concurrent VNF execution aims to reduce the traffic latency of an SFC by executing the VNFs of the SFC concurrently. However, this is only applicable to VNFs that do not modify the traffic packets, such as a flow monitor and an Intrusion Detection System. A copying module duplicates traffic packets and transmits the packets concurrently through the VNFs, and the processed packets are merged by a merging module, as shown in Fig. 21. By processing duplicate packets concurrently and avoiding sequential processing, concurrent VNF execution reduces traffic latency. This technique was found in the works of Agarwal et al. [153], Xie et al. [190], Jia et al. [125], and Li et al. [100].

9.4. Sub-chain Parallelisation

Sub-chain parallelisation aims to reduce traffic latency by parallelising a part of an SFC. Unlike concurrent VNF execution, in which traffic is duplicated and processed concurrently by VNFs, in sub-chain parallelisation, a sub-chain consisting of a sequence of VNFs is duplicated, and the traffic is distributed across the duplicated sub-chains, as shown in Fig. 22. Since traffic does not need to be duplicated, copying and merging modules are not required. This concept was seen in the approach of Wang et al. [145], and Li et al. [88].

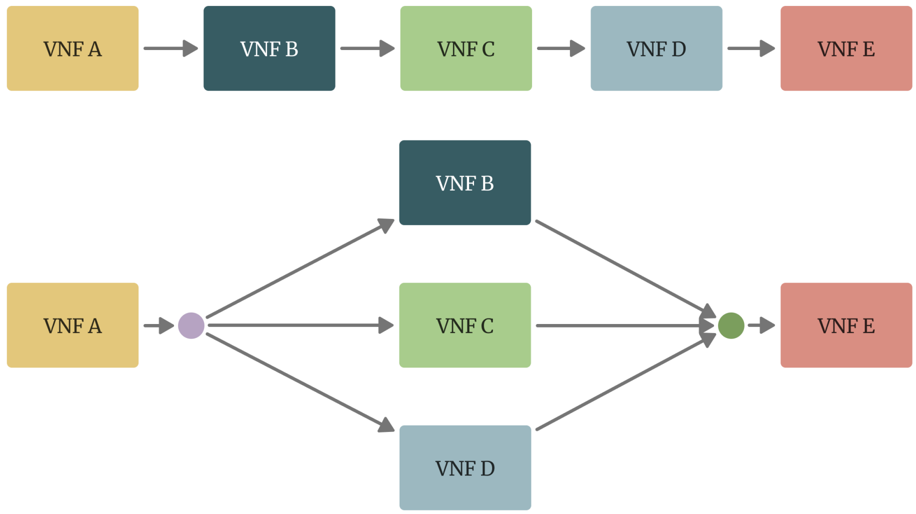

Figure 21: A typical SFC is shown at the top, with concurrent VNF execution for that SFC shown at the bottom. Traffic is duplicated by the copying module, as indicated by the purple circle, and sent in parallel to VNF B, VNF C, and VNF D. The concurrently processed traffic is then merged by the merging module, as shown by the green circle.

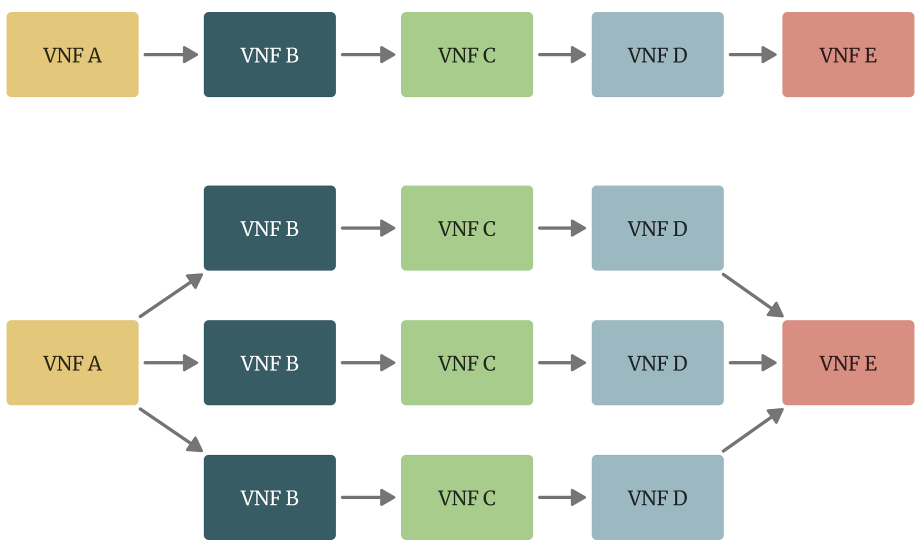

Figure 22: An example of sub-chain parallelisation, with a typical SFC shown at the top. The sub-chain consisting of VNF B, VNF C and VNF D is duplicated, and traffic is distributed across these sub-chains.

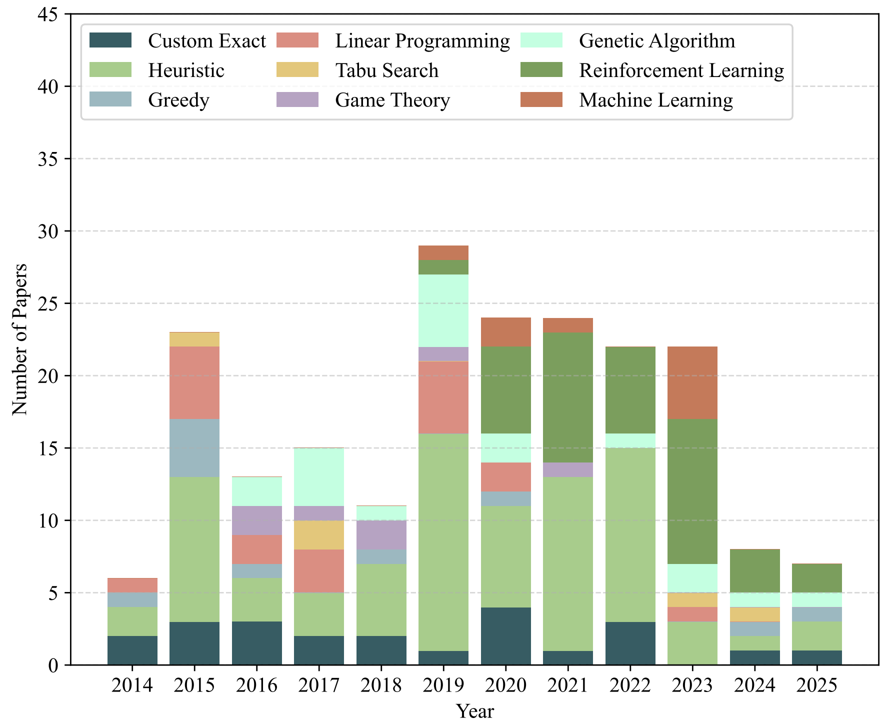

Figure 23: The number of algorithms used in SFC-OCE approaches by year. Only algorithms that were used by more than 2 studies were considered to generate this plot.

9.5. Prediction-based Approaches

Various approaches in the literature used the prediction of the performance of SFCs and the changes to the operating environment to optimise the SFC-OCE problem. This enabled SFC-OCE approaches to adapt to changes pre-emptively. The traffic latency was predicted by Bunyakitanon et al. [96]. Pandey et al. [103, 115] used a Gated Recurrent Unit to predict the resource demands of SFCs. Wang et al. [101] and Mohamad et al. [142] predicted the arrival of SFCRs. Tang et al. [137] developed Digital Twins (DTs) of VNFs that collected resource demand data and monitored VNFs for service failure. These DTs could be used to predict VNF performance for effectively optimising the SFC-OCE problem.

9.6. Reinforcement Learning

The first instance of RL usage in SFC-OCE approaches was seen in 2019, and since then, RL has become a popular algorithm to optimise the SFC-OCE problem, as shown in Fig. 23. Starting from 2020, aside from heuristic algorithms, RL has become the most popular algorithm in the literature. RL-based approaches generally embed VNFs one by one, guided by rewards and penalties to find an optimal SFC embedding.

10. Research Gaps

After analysing the literature on SFC-OCE approaches, in this section, we discuss the research gaps we identified.

10.1. Complete Optimisation of SFC-OCE

The SFC-OCE problem consists of three sub-problems, namely VNF-CC, VNF-EM, and VL-EM, where simultaneous optimisation has led to the best results (Section 4). However, only 10 of the 209 studies optimised all three sub-problems simultaneously, while the majority (89) optimised VNF-EM and VL-EM simultaneously. To support more efficient solutions, more research needs to focus on optimising all three sub-problems simultaneously.

10.2. Explicit Optimisation of VL-EM

Most of the approaches that optimised VL-EM took an implicit approach. An implicit approach is an approach that assumes that servers in a network topology are directly connected via physical links, ignoring network switches, thereby abstracting away the complexity of real computer networks (Section 6.1). Considering explicit approaches, the majority employed a shortest-path algorithm to optimise VL-EM. However, using the shortest path to link servers can lead to traffic congestion on the links in the shortest path [194]. Only 14 studies optimised VL-EM explicitly without using a shortest-path algorithm, making this an underexplored research area.

10.3. High-fidelity Evaluation of SFC-OCE Approaches

78.5% of the approaches performed only numerical evaluation of their approaches. A further 11% used simulators to evaluate their approaches. However, both numerical evaluations and simulations are of lower fidelity (Section 8), meaning 89.5% of the approaches performed low-fidelity evaluations. Evaluating approaches using high-fidelity mechanisms is important to ensure SFC-OCE approaches will have the desired outcome on real networks.

Higher-fidelity evaluations are more challenging to perform than lower-fidelity evaluations, which may explain the unpopularity of higher-fidelity evaluations. Testbeds, which offer the highest fidelity, are costly to set up and maintain, do not scale well, and are time-consuming to configure [239]. PGVTs and emulators, which offer higher fidelity than simulators, cost less, offer more scalability and are easier to set up and configure than testbeds. However, except for one emulator, OpenRASE, none of the PGVTs and emulators natively support evaluating SFC embedding approaches [140]. This means that these PGVTs and emulators require additional development effort to be used for evaluating SFC embedding approaches.

10.4. Dynamic SFC-OCE Approaches

50.7% of the approaches considered a static network environment, where a static set of SFCRs has to be embedded on a static network topology. In the literature, we found three dynamic factors, namely dynamic SFCRs, dynamic traffic and dynamic topology (Section 7.3). Even though 49.3% of the approaches considered one or two of the dynamic factors, none of the approaches considered all three dynamic factors, making this an open research area.

11. Conclusion

We surveyed the existing literature for SFC composition and embedding approaches published between 2014 and 2025, and curated 209 papers after reviewing their title, abstract and content. We then developed an analytical framework to extract data from the curated papers for further analysis. Based on the extracted data, we proposed an improvement to the NFA-RA definition, named SFC-OCE, by decomposing the VNF-FGE sub-problem into two sub-problems, namely VNF-EM and VL-EM, to provide a more detailed classification of the approaches, and disregarding VNF-SCH, as recent advancements have rendered it redundant. Then, we analysed the approaches in terms of their use cases, which included the optimisation objectives, application domains, and physical network topologies, and classified them according to the updated SFC-OCE definition we proposed. We then analysed the algorithms used by the approaches, their scalability and dynamic nature, before analysing the evaluation mechanisms used to evaluate the approaches. Finally, we discussed the emerging trends, such as VNF decomposition, VNF sharing, concurrent VNF execution, sub-chain parallelisation, prediction-based approaches and the use of RL, and identified research gaps, such as the paucity of complete optimisation of SFC-OCE sub-problems, explicit optimisation of the VL-EM sub-problem, high-fidelity evaluation of the approaches and comprehensive dynamic SFC-OCE approaches.

References

[1] K. Pandav, A. G. Te, N. Tomer, S. S. Nair, A. K. Tewari, Leveraging 5G technology for robotic surgery and cancer care, Cancer Reports 5 (2022) e1595. doi:https://doi.org/10.1002/cnr2.1595.

[2] T. Adrian, T. Mancini-Griffoli, The Rise of Digital Money, Annual Review of Financial Economics 13 (2021) 57–77. doi:https://doi.org/10.1146/annurev-financial-101620-063859.

[3] A. Tripathi, B. Parihar, E-Governance challenges and cloud benefits, in: 2011 IEEE International Conference on Computer Science and Automation Engineering, CSAE 2011, volume 1, 2011, pp. 351–354. doi:10.1109/CSAE.2011.5953237.

[4] N.-N. Dao, A.-T. Tran, N. H. Tu, T. T. Thanh, V. N. Q. Bao, S. Cho, A Contemporary Survey on Live Video Streaming from a Computation-Driven Perspective, ACM Comput. Surv. 54 (2022). doi:10.1145/3519552.

[5] P. Imai, P. Harvey, T. Amin, Towards A Truly Autonomous Network, Technical Report, Rakuten Mobile Innovation Studio, 2020.

[6] M. Ghaznavi, N. Shahriar, R. Ahmed, R. Boutaba, Service Function Chaining Simplified, IEEE Journal on Selected Areas in Communications 35 (2016) 2479–2489. doi:10.1109/JSAC.2017.2760178.

[7] NFV, GS NFV 002 - V1.2.1 - Network Functions Virtualisation (NFV); Architectural Framework, 2014.

[8] Software-Defined Networking: Why We Like It and How We Are Building On It, 2013.

[9] J. Gil Herrera, J. F. Botero, Resource Allocation in NFV: A Comprehensive Survey, IEEE Transactions on Network and Service Management 13 (2016) 518–532. doi:10.1109/TNSM.2016.2598420.

[10] K. Kaur, V. Mangat, K. Kumar, A comprehensive survey of service function chain provisioning approaches in SDN and NFV architecture, Computer Science Review 38 (2020) 100298. doi:10.1016/J.COSREV.2020.100298.

[11] F. Schardong, I. Nunes, A. Schaeffer-Filho, NFV Resource Allocation: a Systematic Review and Taxonomy of VNF Forwarding Graph Embedding, Computer Networks 185 (2021) 107726. doi:10.1016/J.COMNET.2020.107726.

[12] B. Zhang, Q. Fan, X. Zhang, Z. Fu, S. Wang, J. Li, Q. Xiong, A survey of VNF forwarding graph embedding in B5G/6G networks, Wireless Networks (2021) 1–24. doi:10.1007/S11276-021-02741-9/TABLES/8.

[13] W. Attaoui, E. Sabir, H. Elbiaze, M. Guizani, VNF and CNF Placement in 5G: Recent Advances and Future Trends, IEEE Transactions on Network and Service Management 20 (2023) 4698–4733. doi:10.1109/TNSM.2023.3264005.

[14] G. L. Santos, D. d. F. Bezerra, Ã. d. S. Rocha, L. Ferreira, A. L. C. Moreira, G. E. Gonçalves, M. V. Marquezini, Ã. Recse, A. Mehta, J. Kelner, D. Sadok, P. T. Endo, Service Function Chain Placement in Distributed Scenarios: A Systematic Review, Journal of Network and Systems Management 30 (2022) 1–39. doi:10.1007/S10922-021-09626-4/TABLES/1.

[15] M. C. Luizelli, L. R. Bays, L. S. Buriol, M. P. Barcellos, L. P. Gaspary, Piecing together the NFV provisioning puzzle: Efficient placement and chaining of virtual network functions, in: 2015 IFIP/IEEE International

Symposium on Integrated Network Management (IM), 2015, pp. 98–106. doi:10.1109/INM.2015.7140281.

[16] G. Bista, E. Caron, A. L. Vion, A Study on Optimizing VNF Software Cost, in: 2020 Global Information Infrastructure and Networking Symposium, GIIS 2020a2020 Global Information Infrastructure and Networking Symposium (GIIS), Institute of Electrical and Electronics Engineers Inc., 2020, pp. 1–4. doi:10.1109/GIIS50753.2020.9248486.

[17] M. B. Chiosi, D. Clarke, P. Willis, A. Reid CenturyLink, J. Feger, M. Bugenhagen, W. Khan, M. Fargano, J. Benitez, U. Michel, H. Damker KDDI, K. Ogaki, T. Matsuzaki NTT, M. Fukui, K. Shimano, D. Delisle, Q. Loudier, C. Kolias, I. Guardini, E. Demaria, R. Minerva, A. Manzalini, D. López, F. Javier Ramón Salguero, F. Ruhl, P. Sen, Network Functions Virtualisation, 2012.

[18] J. M. Halpern, C. Pignataro, Service Function Chaining (SFC) Architecture, RFC 7665, 2015. doi:10.17487/RFC7665.

[19] Z. Kuai, S. Wang, Fair Virtual Network Function Scheduling with Deep Reinforcement Learning, in: 2021 IEEE Global Communications Conference (GLOBECOM), Institute of Electrical and Electronics Engineers Inc., 2021, pp. 1–6. doi:10.1109/GLOBECOM46510.2021.9686006.

[20] Z. Li, L. Wu, X. Zeng, X. Yue, Y. Jing, W. Wu, K. Su, Online Coordinated NFV Resource Allocation via Novel Machine Learning Techniques, IEEE Transactions on Network and Service Management 20 (2023) 563–577. doi:10.1109/TNSM.2022.3205900.

[21] Q. Li, X. Wang, T. Zhao, Y. Wang, Z. Li, L. Rui, An Improved Genetic Algorithm for the Scheduling of Virtual Network Functions, in: 2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS), Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–4. doi:10.23919/APNOMS.2019.8892907.

[22] J. F. Riera, E. Escalona, J. Batallé, E. Grasa, J. A. García-Espín, Virtual network function scheduling: Concept and challenges, in: 2014 International Conference on Smart Communications in Network Technologies (SaCoNeT), IEEE Computer Society, 2014, pp. 1–5. doi:10.1109/SACONET.2014.6867768.

[23] J. F. Riera, X. Hesselbach, E. Escalona, J. A. Garcia-Espin, E. Grasa, On the complex scheduling formulation of virtual network functions over optical networks, in: 2014 16th International Conference on Transparent Optical Networks (ICTON), IEEE Computer Society, 2014, pp. 1–5. doi:10.1109/ICTON.2014.6876564.

[24] L. Wang, Z. Lu, X. Wen, R. Knopp, R. Gupta, Joint Optimization of Service Function Chaining and Resource Allocation in Network Function

Virtualization, IEEE Access 4 (2016) 8084–8094. doi:10.1109/ACCESS.2016.2629278.

[25] J. Aelken, J. Triay, B. Chatras, A. M. de Nicolas, Toward Cloud-Native VNFs: An ETSI NFV Management and Orchestration Standards Approach, IEEE Communications Standards Magazine 8 (2024) 12–19. doi:10.1109/MCOMSTD.0002.2200079.

[26] A. Satpathy, M. Narayan Sahoo, C. Swain, P. Bellavista, M. Guizani, K. Muhammad, S. Bakshi, Virtual Network Embedding: Literature Assessment, Recent Advancements, Opportunities, and Challenges, IEEE Communications Surveys & Tutorials 27 (2025) 3861–3914. doi:10.1109/COMST.2025.3531724.

[27] F. Bachini Lopes, A. E. Schaeffer-Filho, G. L. Nazar, Modular VNF Components Acceleration With FPGA Overlays, IEEE Transactions on Network and Service Management 20 (2023) 846–857. doi:10.1109/TNSM.2022.3211448.

[28] G. S. Niemiec, L. M. S. Batista, A. E. Schaeffer-Filho, G. L. Nazar, A Survey on FPGA Support for the Feasible Execution of Virtualized Network Functions, IEEE Communications Surveys & Tutorials 22 (2020) 504–525. doi:10.1109/COMST.2019.2943690.

[29] D. Harutyunyan, N. Shahriar, R. Boutaba, R. Riggio, Latency and Mobility-Aware Service Function Chain Placement in 5G Networks, IEEE Transactions on Mobile Computing 21 (2022) 1697–1709. doi:10.1109/TMC.2020.3028216.

[30] D. Harutyunyan, N. Shahriar, R. Boutaba, R. Riggio, Latency-Aware Service Function Chain Placement in 5G Mobile Networks, in: Proceedings of the 2019 IEEE Conference on Network Softwarization: Unleashing the Power of Network Softwarization, NetSoft 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 133–141. doi:10.1109/NETSOFT.2019.8806646.

[31] I. Leyva-Pupo, C. Cervelló-Pastor, Efficient solutions to the placement and chaining problem of User Plane Functions in 5G networks, Journal of Network and Computer Applications 197 (2022) 103269. doi:10.1016/J.JNCA.2021.103269.

[32] R. Riggio, A. Bradai, T. Rasheed, J. Schulz-Zander, S. Kuklinski, T. Ahmed, Virtual network functions orchestration in wireless networks, in: Proceedings of the 11th International Conference on Network and Service Management, CNSM 2015, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 108–116. doi:10.1109/CNSM.2015.7367346.

[33] M. Bouet, J. Leguay, V. Conan, Cost-based placement of vDPI functions in NFV infrastructures, in: Proceedings of the 2015 1st IEEE Conference

on Network Softwarization (NetSoft), Institute of Electrical and Electronics Engineers Inc., 2015, pp. 1–9. doi:10.1109/NETSOFT.2015.7116121.

[34] R. Qiu, J. Bao, Y. Li, X. Zhou, L. Liang, H. Tian, Y. Zeng, J. Shi, Virtual network function deployment algorithm based on graph convolution deep reinforcement learning, Journal of Supercomputing 79 (2022) 6849–6870. doi:10.1007/S11227-022-04947-W/TABLES/3.

[35] P. K. Thiruvasagam, A. Chakraborty, A. Mathew, C. S. R. Murthy, Reliable Placement of Service Function Chains and Virtual Monitoring Functions with Minimal Cost in Softwarized 5G Networks, IEEE Transactions on Network and Service Management 18 (2021) 1491–1507. doi:10.1109/TNSM.2021.3056917.

[36] D. Harutyunyan, R. Behravesh, N. Slamnik-Kriještorac, Cost-efficient Placement and Scaling of 5G Core Network and MEC-enabled Application VNFs, in: 2021 IFIP/IEEE International Symposium on Integrated Network Management (IM), 2021, pp. 241–249.

[37] A. Mohamad, H. S. Hassanein, On demonstrating the gain of SFC placement with VNF sharing at the edge, in: 2019 IEEE Global Communications Conference (GLOBECOM), Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–6. doi:10.1109/GLOBECOM38437.2019.9014106.

[38] H. Jin, X. Zhu, C. Zhao, Computation offloading optimization based on probabilistic SFC for mobile online gaming in heterogeneous network, IEEE Access 7 (2019) 52168–52180. doi:10.1109/ACCESS.2019.2909971.

[39] H. Guo, Y. Wang, Z. Li, X. Qiu, H. An, P. Yu, N. Yuan, Cost-aware Placement and Chaining of Service Function Chain with VNF Instance Sharing, Proceedings of IEEE/IFIP Network Operations and Management Symposium 2020: Management in the Age of Softwarization and Artificial Intelligence, NOMS 2020 (2020). doi:10.1109/NOMS47738.2020.9110360.

[40] P. T. A. Quang, A. Bradai, K. D. Singh, G. Picard, R. Riggio, Single and Multi-Domain Adaptive Allocation Algorithms for VNF Forwarding Graph Embedding, IEEE Transactions on Network and Service Management 16 (2019) 98–112. doi:10.1109/TNSM.2018.2876623.

[41] A. Baumgartner, V. S. Reddy, T. Bauschert, Mobile core network virtualization: A model for combined virtual core network function placement and topology optimization, in: Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft), Institute of Electrical and Electronics Engineers Inc., 2015, pp. 1–9. doi:10.1109/NETSOFT.2015.7116162.

[42] A. Baumgartner, V. S. Reddy, T. Bauschert, Combined Virtual Mobile Core Network Function Placement and Topology Optimization with Latency Bounds, in: European Workshop on Software Defined Networks,

EWSDN, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 97–102. doi:10.1109/EWSDN.2015.68.

[43] L. Liu, S. Guo, G. Liu, Y. Yang, Joint Dynamical VNF Placement and SFC Routing in NFV-Enabled SDNs, IEEE Transactions on Network and Service Management 18 (2021) 4263–4276. doi:10.1109/TNSM.2021.3091424.

[44] B. Raddwan, K. AL-Wagih, I. A. Al-Baltah, M. A. Alrshah, M. A. Al-Maqri, Path Mapping Approach for Network Function Virtualization Resource Allocation with Network Function Decomposition Support, Symmetry 2019, Vol. 11, Page 1173 11 (2019) 1173. doi:10.3390/SYM11091173.

[45] M. A. Khoshkholghi, J. Taheri, D. Bhamare, A. Kassler, Optimized Service Chain Placement Using Genetic Algorithm, in: Proceedings of the 2019 IEEE Conference on Network Softwarization: Unleashing the Power of Network Softwarization, NetSoft 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 472–479. doi:10.1109/NETSOFT.2019.8806644.

[46] M. Ghaznavi, A. Khan, N. Shahriar, K. Alsubhi, R. Ahmed, R. Boutaba, Elastic virtual network function placement, in: 2015 IEEE 4th International Conference on Cloud Networking, CloudNet 2015, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 255–260. doi:10.1109/CLOUDNET.2015.7335318.

[47] A. Khatiri, G. Mirjalily, Z. Q. Luo, Balanced resource allocation for VNF service chain provisioning in inter-datacenter elastic optical networks, Computer Networks 203 (2022) 108717. doi:10.1016/J.COMNET.2021.108717.

[48] R. Mohamed, A. Leivadeas, I. Lambadaris, T. Moris, P. Djukic, Fast Resource Allocation for Virtual Network Functions Chain Placement, in: 2022 International Telecommunications Conference (ITC-Egypt), Institute of Electrical and Electronics Engineers Inc., 2022, pp. 1–6. doi:10.1109/ITC-EGYPT55520.2022.9855716.

[49] C. T. Wang, Y. D. Lin, C. C. Wang, Y. C. Lai, Cost minimization in placing service chains for virtualized network functions, International Journal of Communication Systems 33 (2020) e4222. doi:10.1002/DAC.4222.

[50] F. Carpio, S. Dhahri, A. Jukan, VNF placement with replication for Load balancing in NFV networks, in: 2017 IEEE International Conference on Communications (ICC), Institute of Electrical and Electronics Engineers Inc., 2017, pp. 1–6. doi:10.1109/ICC.2017.7996515.

[51] T. Lin, Z. Zhou, M. Tornatore, B. Mukherjee, Optimal Network Function Virtualization Realizing End-to-End Requests, in: 2015 IEEE Global

Communications Conference (GLOBECOM), Institute of Electrical and Electronics Engineers (IEEE), 2015, pp. 1–6. doi:10.1109/GLOCOM.2015.7417447.

[52] R. Cohen, L. Lewin-Eytan, J. S. Naor, D. Raz, Near optimal placement of virtual network functions, in: 2015 IEEE Conference on Computer Communications (INFOCOM), volume 26, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 1346–1354. doi:10.1109/INFOCOM.2015.7218511.

[53] T. Lin, Z. Zhou, M. Tornatore, B. Mukherjee, Demand-Aware Network Function Placement, Journal of Lightwave Technology 34 (2016) 2590–2600. doi:10.1109/JLT.2016.2535401.

[54] M. F. Bari, S. R. Chowdhury, R. Ahmed, R. Boutaba, On orchestrating virtual network functions, in: Proceedings of the 11th International Conference on Network and Service Management, CNSM 2015, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 50–56. doi:10.1109/CNSM.2015.7367338.

[55] A. Allahvirdi, S. Yousefi, A. Asgharian Sardroud, Placement of dynamic service function chains in partially VNF-enabled networks, Telecommunication Systems 81 (2022) 225–240. doi:10.1007/S11235-022-00939-6/FIGURES/7.

[56] T. V. Doan, G. T. Nguyen, M. Reisslein, F. H. Fitzek, SAP: Subchain-Aware NFV Service Placement in Mobile Edge Cloud, IEEE Transactions on Network and Service Management 20 (2023) 319–341. doi:10.1109/TNSM.2022.3201388.

[57] J. Pei, P. Hong, M. Pan, J. Liu, J. Zhou, Optimal VNF Placement via Deep Reinforcement Learning in SDN/NFV-Enabled Networks, IEEE Journal on Selected Areas in Communications 38 (2020) 263–278. doi:10.1109/JSAC.2019.2959181.

[58] G. L. Santos, P. T. Endo, D. Sadok, J. Kelner, SPIDER: An availability-aware framework for the service function chain placement in distributed scenarios, Software: Practice and Experience 53 (2023) 413–437. doi:10.1002/SPE.3154.

[59] X. Shang, Z. Liu, Y. Yang, Online Service Function Chain Placement for Cost-Effectiveness and Network Congestion Control, IEEE Transactions on Computers 71 (2022) 27–39. doi:10.1109/TC.2020.3035991.

[60] X. Shang, Z. Liu, Y. Yang, Network congestion-aware online service function chain placement and load balancing, in: Proceedings of the 48th International Conference on Parallel Processing, Association for Computing Machinery, 2019, pp. 1–10. doi:10.1145/3337821.3337850.

[61] A. L. Ã. Battisti, E. L. C. Macedo, M. I. P. Josué, H. Barbalho, F. C. Delicato, D. C. Muchaluat-Saade, P. F. Pires, D. P. d. Mattos, A. C. B. d. Oliveira, A Novel Strategy for VNF Placement in Edge Computing Environments, Future Internet 14 (2022) 361. doi:10.3390/fi14120361.

[62] S. Guo, Y. Dai, S. Xu, X. Qiu, F. Qi, Trusted Cloud-Edge Network Resource Management: DRL-Driven Service Function Chain Orchestration for IoT, IEEE Internet of Things Journal 7 (2020) 6010–6022. doi:10.1109/JIOT.2019.2951593.

[63] A. Mohamad, H. S. Hassanein, PSVShare: A Priority-based SFC placement with VNF Sharing, in: 2020 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Institute of Electrical and Electronics Engineers Inc., 2020, pp. 25–30. doi:10.1109/NFV-SDN50289.2020.9289837.

[64] G. L. Santos, P. T. Endo, T. Lynn, D. Sadok, J. Kelner, A reinforcement learning-based approach for availability-aware service function chain placement in large-scale networks, Future Generation Computer Systems 136 (2022) 93–109. doi:10.1016/J.FUTURE.2022.05.021.

[65] M. A. Abdelaal, G. A. Ebrahim, W. R. Anis, Efficient Placement of Service Function Chains in Cloud Computing Environments, Electronics 10 (2021) 323. doi:10.3390/electronics10030323.

[66] A. Khatiri, G. Mirjalily, A cost-efficient, load-balanced and fragmentation-aware approach for deployment of VNF service chains in Elastic Optical Networks, Computer Communications 188 (2022) 156–166. doi:10.1016/J.COMCOM.2022.03.005.

[67] Y. Zhang, F. Zhang, S. Tong, A. Rezaeipanah, A dynamic planning model for deploying service functions chain in fog-cloud computing, Journal of King Saud University - Computer and Information Sciences 34 (2022) 7948–7960. doi:10.1016/J.JKSUCI.2022.07.012.

[68] O. Almurshed, O. Rana, K. Chard, Greedy Nominator Heuristic: Virtual function placement on fog resources, Concurrency and Computation: Practice and Experience 34 (2022) e6765. doi:10.1002/CPE.6765.

[69] N. Toumi, O. Bernier, D. E. Meddour, A. Ksentini, On Using Physical Programming for Multi-Domain SFC Placement With Limited Visibility, IEEE Transactions on Cloud Computing 10 (2022) 2787–2803. doi:10.1109/TCC.2020.3046997.

[70] N. Toumi, M. Bagaa, A. Ksentini, Hierarchical multi-agent deep reinforcement learning for SFC placement on multiple domains, in: Conference on Local Computer Networks, LCN, volume 2021-Octob, IEEE Computer Society, 2021, pp. 299–304. doi:10.1109/LCN52139.2021.9524980.

[71] R. Mijumbi, J. Serrat, J. L. Gorricho, J. Rubio-Loyola, S. Davy, Server placement and assignment in virtualized radio access networks, in: Proceedings of the 11th International Conference on Network and Service Management, CNSM 2015, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 398–401. doi:10.1109/CNSM.2015.7367390.

[72] Y. Mao, X. Shang, Y. Yang, Provably Efficient Algorithms for Traffic-sensitive SFC Placement and Flow Routing, in: IEEE INFOCOM 2022 - IEEE Conference on Computer Communications, volume 2022-May, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 950–959. doi:10.1109/INFOCOM48880.2022.9796690.

[73] L. Gao, Y. Chen, B. Tang, Service Function Chain Placement in Cloud Data Center Networks: A Cooperative Multi-agent Reinforcement Learning Approach, Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST 457 LNICST (2022) 291–309. doi:10.1007/978-3-031-23141-4{_}22/TABLES/2.

[74] M. A. Khoshkholghi, M. Gokan Khan, K. Alizadeh Noghani, J. Taheri, D. Bhamare, A. Kassler, Z. Xiang, S. Deng, X. Yang, Service Function Chain Placement for Joint Cost and Latency Optimization, Mobile Networks and Applications 25 (2020) 2191–2205. doi:10.1007/S11036-020-01661-W/TABLES/5.

[75] P. D. Nguyen, L. B. Le, Joint Computation Offloading, SFC Placement, and Resource Allocation for Multi-Site MEC Systems, in: 2020 IEEE Wireless Communications and Networking Conference (WCNC), volume 2020-May, Institute of Electrical and Electronics Engineers Inc., 2020, pp. 1–6. doi:10.1109/WCNC45663.2020.9120597.

[76] G. Dandachi, S. Cerf, Y. Hadjadj-Aoul, A. Outtagarts, E. Rutten, A robust control-theory-based exploration strategy in deep reinforcement learning for virtual network embedding, Computer Networks 218 (2022) 109366. doi:10.1016/J.COMNET.2022.109366.

[77] J. Liu, W. Lu, F. Zhou, P. Lu, Z. Zhu, On Dynamic service function chain deployment and readjustment, IEEE Transactions on Network and Service Management 14 (2017) 543–553. doi:10.1109/TNSM.2017.2711610.

[78] H. A. Shah, L. Zhao, Multiagent Deep-Reinforcement-Learning-Based Virtual Resource Allocation through Network Function Virtualization in Internet of Things, IEEE Internet of Things Journal 8 (2021) 3410–3421. doi:10.1109/JIOT.2020.3022572.

[79] Z. Zeng, Z. Xia, X. Zhang, Y. He, SFC Design and VNF Placement Based on Traffic Volume Scaling and VNF Dependency in 5G Networks, Computer Modeling in Engineering & Sciences 134 (2022). doi:10.32604/cmes.2022.021648.

[80] S. Sahhaf, W. Tavernier, M. Rost, S. Schmid, D. Colle, M. Pickavet, P. Demeester, Network service chaining with optimized network function embedding supporting service decompositions, Comput. Networks 93 (2015) 492–505. doi:10.1016/J.COMNET.2015.09.035.

[81] J. Li, W. Shi, Q. Ye, W. Zhuang, X. Shen, X. Li, Online Joint VNF Chain Composition and Embedding for 5G Networks, in: 2018 IEEE Global Communications Conference (GLOBECOM), Institute of Electrical and Electronics Engineers Inc., 2018, pp. 1–6. doi:10.1109/GLOCOM.2018.8647700.

[82] Y. Mao, X. Shang, Y. Yang, Joint Resource Management and Flow Scheduling for SFC Deployment in Hybrid Edge-and-Cloud Network, in: IEEE INFOCOM 2022 - IEEE Conference on Computer Communications, volume 2022-May, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 170–179. doi:10.1109/INFOCOM48880.2022.9796884.

[83] A. Azhdari, A. Ebrahimzadeh, S. N. Afrasiabi, R. Szabó, C. Mouradian, W. Li, R. H. Glitho, Cost-Aware Topological Decomposition of Virtual Network Function Forwarding Graphs, in: IEEE Global Communications Conference, GLOBECOM, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 5384–5390. doi:10.1109/GLOBECOM54140.2023.10437838.

[84] H. Li, L. Wang, X. Wen, Z. Lu, L. Ma, Constructing Service Function Chain Test Database: An Optimal Modeling Approach for Coordinated Resource Allocation, IEEE Access 6 (2017) 17595–17605. doi:10.1109/ACCESS.2017.2780991.

[85] T. M. Nguyen, S. Fdida, T. M. Pham, A comprehensive resource management and placement for network function virtualization, in: 2017 IEEE Conference on Network Softwarization (NetSoft), Institute of Electrical and Electronics Engineers Inc., 2017, pp. 1–9. doi:10.1109/NETSOFT.2017.8004197.

[86] V. Fulber-Garcia, J. Flauzino, G. Venâncio, A. Huff, E. P. Duarte, Breaking the Limits: Bio-Inspired SFC Deployment across Multiple Domains, Clouds and Orchestrators, in: 2024 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Institute of Electrical and Electronics Engineers Inc., 2024, pp. 1–6. doi:10.1109/NFV-SDN61811.2024.10807470.

[87] B. Yu, W. Zheng, X. Wen, Z. Lu, L. Wang, L. Ma, Dynamic Resource Orchestration of Service Function Chaining in Network Function Virtualizations, in: K. Long, V. C. M. Leung, H. Zhang, Z. Feng, Y. Li, Z. Zhang (Eds.), 5G for Future Wireless Networks, Springer International Publishing, Cham, 2018, pp. 132–145.

[88] H. Li, L. Wang, X. Wen, Z. Lu, J. Li, MSV: An Algorithm for Coordinated Resource Allocation in Network Function Virtualization, IEEE Access 6 (2018) 76876–76888. doi:10.1109/ACCESS.2018.2882255.

[89] L. Xu, H. Hu, Y. Liu, SFCSim: a network function virtualization resource allocation simulation platform, Cluster Computing 26 (2023) 423–436. doi:10.1007/S10586-022-03670-8/FIGURES/13.

[90] S. N. Afrasiabi, A. Ebrahimzadeh, A. Azhdari, C. Mouradian, W. Li, R. Szabó, R. H. Glitho, Joint VNF Decomposition and Migration for Cost-Efficient VNF Forwarding Graph Embedding, in: IEEE Global Communications Conference, GLOBECOM, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 2863–2869. doi:10.1109/GLOBECOM54140.2023.10437348.

[91] S. Malektaji, M. Rayani, A. Ebrahimzadeh, V. M. Raee, H. Elbiaze, R. H. Glitho, Dynamic Joint VNF Forwarding Graph Composition and Embedding: A Deep Reinforcement Learning Framework, IEEE Transactions on Network and Service Management 20 (2023) 4615–4633. doi:10.1109/TNSM.2023.3258192.

[92] A. Hentati, A. Ebrahimzadeh, R. H. Glitho, F. Belqasmi, R. Mizouni, Deterministic and Dynamic Joint Placement and Scheduling of VNF-FGs for Remote Robotic Surgery, IEEE Transactions on Network and Service Management (2025). doi:10.1109/TNSM.2025.3539183.

[93] A. Azhdari, A. Ebrahimzadeh, C. Mouradian, R. Szabó, R. H. Glitho, Cost-Aware VNF Decomposition for VNF Forwarding Graph Embedding, IEEE Transactions on Network and Service Management (2025) 1–1. doi:10.1109/TNSM.2025.3614632.

[94] E. Sargolzaei, M. Rasti, S. Khorsandi, Topology and Energy Aware Approximate Algorithm for QoS-based Resource Slicing in 5G Core Networks, IEEE Access PP (2025) 1–1. doi:10.1109/access.2025.3616851.

[95] S. N. Afrasiabi, A. Ebrahimzadeh, N. Promwongsa, C. Mouradian, W. Li, A. Recse, R. Szabo, R. H. Glitho, Cost-Efficient Cluster Migration of VNFs for Service Function Chain Embedding, IEEE Transactions on Network and Service Management 21 (2024) 979–993. doi:10.1109/TNSM.2023.3287757.

[96] M. Bunyakitanon, A. P. da Silva, X. Vasilakos, R. Nejabati, D. Simeonidou, Auto-3P: An autonomous VNF performance prediction & placement framework based on machine learning, Computer Networks 181 (2020) 107433. doi:10.1016/J.COMNET.2020.107433.

[97] B. Martini, F. Paganelli, P. Cappanera, S. Turchi, P. Castoldi, Latency-aware composition of Virtual Functions in 5G, in: Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft), Institute

of Electrical and Electronics Engineers Inc., 2015, pp. 1–6. doi:10.1109/NETSOFT.2015.7116188.

[98] T. Subramanya, D. Harutyunyan, R. Riggio, Machine learning-driven service function chain placement and scaling in MEC-enabled 5G networks, Computer Networks 166 (2020) 106980. doi:10.1016/J.COMNET.2019.106980.

[99] D. M. Manias, I. Shaer, J. Naoum-Sawaya, A. Shami, Robust and Reliable SFC Placement in Resource-Constrained Multi-Tenant MEC-Enabled Networks, IEEE Transactions on Network and Service Management (2023) 1–1. doi:10.1109/TNSM.2023.3293027.

[100] H. P. Li, M. E. Kordi, DSPPV: Dynamic service function chains placement with parallelized virtual network functions in mobile edge computing, Internet of Things 22 (2023) 100733. doi:10.1016/J.IOT.2023.100733.

[101] X. Wang, H. Xing, F. Song, S. Luo, P. Dai, Dynamic Multicast-oriented Virtual Network Function Placement with SFC Request Prediction, in: 2022 14th International Conference on Communication Software and Networks, ICCSN 2022, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 81–88. doi:10.1109/ICCSN55126.2022.9817590.

[102] P. Zou, S. Yang, T. Zhang, S. Wei, Risk-Aware SFC Placement Method in Edge Cloud Environment, in: Emerging Networking Architecture and Technologies, volume 1696 CCIS, Springer Science and Business Media Deutschland GmbH, 2023, pp. 105–116. doi:10.1007/978-981-19-9697-9{_}9.

[103] S. Pandey, M. Choi, J. H. Yoo, J. W. K. Hong, RNN-EdgeQL: An autoscaling and placement approach for SFC, International Journal of Network Management 33 (2023) e2213. doi:10.1002/NEM.2213.

[104] Y. Liu, X. Shang, Y. Yang, Joint SFC Deployment and Resource Management in Heterogeneous Edge for Latency Minimization, IEEE Transactions on Parallel and Distributed Systems 32 (2021) 2131–2143. doi:10.1109/TPDS.2021.3062341.

[105] S. Pandey, T. Van Nguyen, J. H. Yoo, J. W. K. Hong, EdgeDQN: Multiple SFC Placement in Edge Computing Environment, in: Proceedings of the 2021 17th International Conference on Network and Service Management: Smart Management for Future Networks and Services, CNSM 2021, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 301–309. doi:10.23919/CNSM52442.2021.9615587.

[106] M. Wang, B. Cheng, W. Feng, J. Chen, An efficient service function chain placement algorithm in a MEC-NFV environment, in: 2019 IEEE Global Communications Conference (GLOBECOM), Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–6. doi:10.1109/GLOBECOM38437.2019.9013235.

[107] R. Riggio, T. Rasheed, R. Narayanan, Virtual network functions orchestration in enterprise WLANs, Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management, IM 2015 (2015) 1220–1225. doi:10.1109/INM.2015.7140470.

[108] F. Zahedi, M. Mollahoseini Ardakani, A. Heidary-Sharifabad, DSPVR: dynamic SFC placement with VNF reuse in Fog-Cloud Computing using Deep Reinforcement Learning, Journal of Ambient Intelligence and Humanized Computing 14 (2023) 3981–3994. doi:10.1007/S12652-022-04465-W/FIGURES/7.

[109] J. Santos, T. Wauters, B. Volckaert, F. De Turck, Towards delay-aware container-based Service Function Chaining in Fog Computing, Proceedings of IEEE/IFIP Network Operations and Management Symposium 2020: Management in the Age of Softwarization and Artificial Intelligence, NOMS 2020 (2020). doi:10.1109/NOMS47738.2020.9110376.

[110] P. Zhang, Y. Zhang, N. Kumar, M. Guizani, Dynamic SFC Embedding Algorithm Assisted by Federated Learning in Space-Air-Ground-Integrated Network Resource Allocation Scenario, IEEE Internet of Things Journal 10 (2023) 9308–9318. doi:10.1109/JIOT.2022.3222200.

[111] M. Bekhit, A. Fathalla, E. Eldesouky, A. Salah, Multi-objective VNF Placement Optimization with NSGA-III, in: K. Daimi, A. Al Sadoon (Eds.), Proceedings of the 2023 International Conference on Advances in Computing Research (ACR'23), Springer Nature Switzerland, Cham, 2023, pp. 481–493. doi:10.1007/978-3-031-33743-7{_}39.

[112] Z. Sharif, M. B. Jasser, K. L. A. Yau, A. Amphawan, Towards Latency Aware Multi-joint Optimization Method for VNF Placement and SFC Routing Via Swarm Intelligence, in: ICCSCE 2022 - Proceedings: 2022 12th IEEE International Conference on Control System, Computing and Engineering, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 25–30. doi:10.1109/ICCSCE54767.2022.9935663.

[113] N. Toumi, M. Bagaa, A. Ksentini, On using Deep Reinforcement Learning for Multi-Domain SFC placement, in: 021 IEEE Global Communications Conference (GLOBECOM), Institute of Electrical and Electronics Engineers Inc., 2021, pp. 1–6. doi:10.1109/GLOBECOM46510.2021.9685367.

[114] M. M. Erbati, G. Schiele, Application-and reliability-Aware service function chaining to support low-latency applications in an NFV-enabled network, in: 2021 IEEE Conference on Network Function Virtualization and Software Defined Networks, NFV-SDN 2021 - Proceedings, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 120–123. doi:10.1109/NFV-SDN53031.2021.9665118.

[115] S. Pandey, J. W. K. Hong, J. H. Yoo, GRU and EdgeQ-Learning based Traffic Prediction and Scaling of SFC, in: Proceedings of the 2021 IEEE

Conference on Network Softwarization: Accelerating Network Softwarization in the Cognitive Age, NetSoft 2021, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 124–132. doi:10.1109/NETSOFT51509.2021.9492570.

[116] M. Gamal, S. Jafarizadeh, M. Abolhasan, J. Lipman, W. Ni, Mapping and scheduling for non-uniform arrival of virtual network function (VNF) requests, in: 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall), volume 2019-Septe, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–6. doi:10.1109/VTCFALL.2019.8891197.

[117] D. M. Manias, M. Jammal, H. Hawilo, A. Shami, P. Heidari, A. Larabi, R. Brunner, Machine learning for performance-aware virtual network function placement, in: 2019 IEEE Global Communications Conference (GLOBECOM), Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–6. doi:10.1109/GLOBECOM38437.2019.9013246.

[118] M. Gamal, M. Abolhasan, S. Jafarizadeh, J. Lipman, W. Ni, Mapping and Scheduling of Virtual Network Functions using Multi Objective Optimization Algorithm, 2019 19th International Symposium on Communications and Information Technologies, ISCIT 2019 (2019) 328–333. doi:10.1109/ISCIT.2019.8905113.

[119] L. Qu, C. Assi, K. Shaban, Delay-Aware Scheduling and Resource Optimization with Network Function Virtualization, IEEE Transactions on Communications 64 (2016) 3746–3758. doi:10.1109/TCOMM.2016.2580150.

[120] R. Bruschi, A. Carrega, F. Davoli, A Game for Energy-Aware Allocation of Virtualized Network Functions, Journal of Electrical and Computer Engineering 2016 (2016). doi:10.1155/2016/4067186.

[121] R. Mijumbi, J. Serrat, J. L. Gorricho, N. Bouten, F. De Turck, S. Davy, Design and evaluation of algorithms for mapping and scheduling of virtual network functions, in: Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft), Institute of Electrical and Electronics Engineers Inc., 2015, pp. 1–9. doi:10.1109/NETSOFT.2015.7116120.

[122] C. Pham, N. H. Tran, C. S. Hong, Virtual Network Function Scheduling: A Matching Game Approach, IEEE Communications Letters 22 (2018) 69–72. doi:10.1109/LCOMM.2017.2747509.

[123] H. A. Alameddine, S. Sebbah, C. Assi, On the interplay between network function mapping and scheduling in VNF-based networks: A column generation approach, IEEE Transactions on Network and Service Management 14 (2017) 860–874. doi:10.1109/TNSM.2017.2757266.

[124] J. F. Riera, X. Hesselbach, M. Zotkiewicz, M. Szostak, J. F. Botero, Modelling the NFV forwarding graph for an optimal network service

deployment, in: 2015 17th International Conference on Transparent Optical Networks (ICTON), volume 2015-Augus, IEEE, 2015, pp. 1–4. doi:10.1109/ICTON.2015.7193483.

[125] J. Jia, J. Hua, Dynamic SFC placement with parallelized VNFs in Data Center Networks: A DRL-based approach, ICT Express (2023). doi:10.1016/J.ICTE.2023.07.005.

[126] Y. Xu, V. P. Kafle, An Availability-Enhanced Service Function Chain Placement Scheme in Network Function Virtualization, Journal of Sensor and Actuator Networks 2019, Vol. 8, Page 34 8 (2019) 34. doi:10.3390/JSAN8020034.

[127] S. Mehraghdam, M. Keller, H. Karl, Specifying and placing chains of virtual network functions, in: 2014 IEEE 3rd International Conference on Cloud Networking, CloudNet 2014, Institute of Electrical and Electronics Engineers Inc., 2014, pp. 7–13. doi:10.1109/CLOUDNET.2014.6968961.

[128] N. Ma, J. Zhang, T. Huang, A model based on genetic algorithm for service chain resource allocation in NFV, in: 2017 3rd IEEE International Conference on Computer and Communications, ICCC 2017, volume 2018-Janua, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 607–611. doi:10.1109/COMPCOMM.2017.8322616.

[129] H. Kim, Virtual resources allocation scheme in ICT converged networks, Lecture Notes in Electrical Engineering 424 (2017) 757–762. doi:10.1007/978-981-10-4154-9{_}87/FIGURES/5.

[130] B. Tahmasebi, M. A. Maddah-Ali, S. Parsaeefard, B. H. Khalaj, Optimum Transmission Delay for Function Computation in NFV-Based Networks: The Role of Network Coding and Redundant Computing, IEEE Journal on Selected Areas in Communications 36 (2018) 2233–2245. doi:10.1109/JSAC.2018.2869952.

[131] Q. Yuan, H. Tang, W. You, X. Wang, Y. Zhao, Virtual network function scheduling via multilayer encoding genetic algorithm with distributed bandwidth allocation, Science China Information Sciences 61 (2018) 1–19. doi:10.1007/S11432-017-9357-7/METRICS.

[132] H. Kim, Revised Virtual Resources Allocation Scheme in Network Function Virtualization (NFV) Enabled Networks, in: K. J. Kim, N. Joukov (Eds.), Mobile and Wireless Technologies 2017, Springer Singapore, Singapore, 2018, pp. 417–423.

[133] Y. Bi, M. Bunyakitanon, N. Uniyal, A. Bravalheri, A. Muqaddas, R. Nejabati, D. Simeonidou, Distributed online resource allocation using congestion game for 5G virtual network services, in: 2019 IEEE Global Communications Conference (GLOBECOM), 2019, pp. 1–7. doi:10.1109/GLOBECOM38437.2019.9013245.

[134] Y. Bi, C. Colman-Meixner, R. Wang, F. Meng, R. Nejabati, D. Simeonidou, Resource Allocation for Ultra-Low Latency Virtual Network Services in Hierarchical 5G Network, in: IEEE International Conference on Communications, volume 2019-May, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–7. doi:10.1109/ICC.2019.8761272.

[135] X. Fu, F. R. Yu, J. Wang, Q. Qi, J. Liao, Dynamic Service Function Chain Embedding for NFV-Enabled IoT: A Deep Reinforcement Learning Approach, IEEE Transactions on Wireless Communications 19 (2020) 507–519. doi:10.1109/TWC.2019.2946797.

[136] M. Escheikh, W. Taktak, Online QoS/QoE-Driven SFC Orchestration Leveraging a DRL Approach in SDN/NFV Enabled Networks, Wireless Personal Communications 137 (2024) 1511–1538. doi:10.1007/S11277-024-11389-5/FIGURES/10.

[137] L. Tang, W. Wen, J. Li, D. Fang, L. Li, Q. Chen, Digital-Twin-Assisted VNF Mapping and Scheduling in SDN/NFV-Enabled Industrial IoT, IEEE Internet of Things Journal 11 (2024) 18516–18533. doi:10.1109/JIOT.2024.3364152.

[138] X. Liu, Y. Hou, H. Liu, Discrete spider monkey optimization algorithm for latency-sensitive VNF deployment and resource allocation, International Journal of Communication Systems 37 (2024) e5640. doi:10.1002/DAC.5640.

[139] W. Wei, Q. Wang, H. Gu, D. Zheng, N. Zhang, C. Wu, An Adaptive Service Function Chains Mapping With Multi-Task Deep Reinforcement Learning, IEEE Transactions on Network Science and Engineering 12 (2025) 3093–3107. doi:10.1109/TNSE.2025.3556390.

[140] T. Krishnamohan, P. Harvey, OpenRASE: Service Function Chain Emulation, in: International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 2025, pp. 1–6. doi:10.23919/SoftCOM66362.2025.11197454.

[141] Y. T. Woldeyohannes, A. Mohammadkhan, K. K. Ramakrishnan, Y. Jiang, ClusPR: Balancing Multiple Objectives at Scale for NFV Resource Allocation, IEEE Transactions on Network and Service Management 15 (2018) 1307–1321. doi:10.1109/TNSM.2018.2870733.

[142] A. Mohamad, H. S. Hassanein, Prediction-based SFC Placement with VNF Sharing at the Edge, in: Conference on Local Computer Networks, LCN, IEEE Computer Society, 2022, pp. 26–33. doi:10.1109/LCN53696.2022.9843704.

[143] J. J. Alves Esteves, A. Boubendir, F. Guillemin, P. Sens, A Heuristically Assisted Deep Reinforcement Learning Approach for Network Slice Placement, IEEE Transactions on Network and Service Management 19 (2022) 4794–4806. doi:10.1109/TNSM.2021.3132103.

[144] M. A. Abdelaal, G. A. Ebrahim, W. R. Anis, High Availability Deployment of Virtual Network Function Forwarding Graph in Cloud Computing Environments, IEEE Access 9 (2021) 53861–53884. doi:10.1109/ACCESS.2021.3068342.

[145] M. Wang, B. Cheng, S. Wang, J. Chen, Availability- And Traffic-Aware Placement of Parallelized SFC in Data Center Networks, IEEE Transactions on Network and Service Management 18 (2021) 182–194. doi:10.1109/TNSM.2021.3051903.

[146] M. Niu, Q. Han, B. Cheng, M. Wang, Z. Xu, W. Gu, S. Zhang, J. Chen, HARS: A High-Available and Resource-Saving Service Function Chain Placement Approach in Data Center Networks, IEEE Transactions on Network and Service Management 19 (2022) 829–847. doi:10.1109/TNSM.2022.3145103.

[147] W. Rankothge, F. Le, A. Russo, J. Lobo, Optimizing Resource Allocation for Virtualized Network Functions in a Cloud Center Using Genetic Algorithms, IEEE Transactions on Network and Service Management 14 (2017) 343–356. doi:10.1109/TNSM.2017.2686979.

[148] X. Yin, B. Cheng, M. Wang, J. Chen, Availability-Aware Service Function Chain Placement in Mobile Edge Computing, in: 2020 IEEE World Congress on Services, SERVICES 2020, Institute of Electrical and Electronics Engineers Inc., 2020, pp. 69–74. doi:10.1109/SERVICES48979.2020.00028.

[149] M. Wang, B. Cheng, J. Chen, An efficient service function chaining placement algorithm in mobile edge computing, ACM Transactions on Internet Technology 20 (2020). doi:10.1145/3388241.

[150] Y. Yue, B. Cheng, X. Liu, M. Wang, B. Li, J. Chen, Resource Optimization and Delay Guarantee Virtual Network Function Placement for Mapping SFC Requests in Cloud Networks, IEEE Transactions on Network and Service Management 18 (2021) 1508–1523. doi:10.1109/TNSM.2021.3058656.

[151] M. Wang, B. Cheng, J. Chen, Joint Availability- And Traffic-aware Placement of Parallelized Service Chain in NFV-enabled Data Center, in: 2020 IEEE 13th International Conference on Web Services, ICWS 2020, Institute of Electrical and Electronics Engineers Inc., 2020, pp. 216–223. doi:10.1109/ICWS49710.2020.00035.

[152] Y. Yue, B. Cheng, X. Liu, Resource optimization and delay-aware virtual network function placement for mapping sfc requests in NFV-enabled Networks, in: IEEE International Conference on Cloud Computing, CLOUD, volume 2020-Octob, IEEE Computer Society, 2020, pp. 267–274. doi:10.1109/CLOUD49709.2020.00046.

[153] S. Agarwal, V. R. Chintapalli, B. R. Tamma, FlexSFC: Flexible Resource Allocation and VNF Parallelism for Improved SFC Placement, in: Proceedings of the 2022 IEEE International Conference on Network Softwarization: Network Softwarization Coming of Age: New Challenges and Opportunities, NetSoft 2022, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 302–306. doi:10.1109/NETSOFT54395.2022.9844084.

[154] A. Basta, W. Kellerer, M. Hoffmann, H. J. Morper, K. Hoffmann, Applying NFV and SDN to LTE mobile core gateways; the functions placement problem, AllThingsCellular 2014 - Proceedings of the 4th ACM Workshop on All Things Cellular: Operations, Applications, and Challenges (2014) 33–38. doi:10.1145/2627585.2627592.

[155] M. Niu, B. Cheng, J. L. Chen, GPSO: A graph-based heuristic algorithm for service function chain placement in data center networks, in: 2020 IEEE 13th International Conference on Services Computing, SCC 2020, Institute of Electrical and Electronics Engineers Inc., 2020, pp. 256–263. doi:10.1109/SCC49832.2020.00041.

[156] M. Hamann, M. Fischer, Path-Based Optimization of NFV-Resource Allocation in SDN Networks, in: IEEE International Conference on Communications, volume 2019-May, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–6. doi:10.1109/ICC.2019.8761738.

[157] H. Chai, J. Zhang, Z. Wang, J. Shi, T. Huang, A Parallel Placement Approach for Service Function Chain Using Deep Reinforcement Learning, in: 2019 IEEE 5th International Conference on Computer and Communications, ICCC 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 2123–2128. doi:10.1109/ICCC47050.2019.9064448.

[158] W. Wang, P. Hong, D. Lee, J. Pei, L. Bo, Virtual network forwarding graph embedding based on Tabu Search, in: 2017 9th International Conference on Wireless Communications and Signal Processing, WCSP 2017 - Proceedings, volume 2017-Janua, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 1–6. doi:10.1109/WCSP.2017.8171072.

[159] J. Soares, S. Sargento, Optimizing the embedding of virtualized cloud network infrastructures across multiple domains, in: 2015 IEEE International Conference on Communications (ICC), 2015, pp. 442–447. doi:10.1109/ICC.2015.7248361.

[160] T. M. Nguyen, M. Minoux, S. Fdida, Optimizing resource utilization in NFV dynamic systems: New exact and heuristic approaches, Computer Networks 148 (2019) 129–141. doi:10.1016/J.COMNET.2018.11.009.

[161] J. J. A. Esteves, A. Boubendir, F. Guillemin, P. Sens, Location-based data model for optimized network slice placement, in: Proceedings of the 2020 IEEE Conference on Network Softwarization: Bridg-

ing the Gap Between AI and Network Softwarization, NetSoft 2020, Institute of Electrical and Electronics Engineers Inc., 2020, pp. 404–412. doi:10.1109/NETSOFT48620.2020.9165427.

[162] L. Qu, C. Assi, M. J. Khabbaz, Y. Ye, Reliability-Aware Service Function Chaining with Function Decomposition and Multipath Routing, IEEE Transactions on Network and Service Management 17 (2020) 835–848. doi:10.1109/TNSM.2019.2961153.

[163] B. Addis, D. Belabed, M. Bouet, S. Secci, Virtual network functions placement and routing optimization, in: 2015 IEEE 4th International Conference on Cloud Networking, CloudNet 2015, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 171–177. doi:10.1109/CLOUDNET.2015.7335301.

[164] I. Jang, S. Choo, M. Kim, S. Pack, M. K. Shin, Optimal network resource utilization in service function chaining, in: 2016 IEEE NetSoft Conference and Workshops (NetSoft), Institute of Electrical and Electronics Engineers Inc., 2016, pp. 11–14. doi:10.1109/NETSOFT.2016.7502433.

[165] A. Mohammadkhan, S. Ghapani, G. Liu, W. Zhang, K. K. Ramakrishnan, T. Wood, Virtual function placement and traffic steering in flexible and dynamic software defined networks, IEEE Workshop on Local and Metropolitan Area Networks 2015-May (2015). doi:10.1109/LANMAN.2015.7114738.

[166] A. Gupta, M. F. Habib, P. Chowdhury, M. Tornatore, B. Mukherjee, On service chaining using Virtual Network Functions in Network-enabled Cloud systems, International Symposium on Advanced Networks and Telecommunication Systems, ANTS 2016-Febru (2015). doi:10.1109/ANTS.2015.7413643.

[167] A. F. Ocampo, J. Gil-Herrera, P. H. Isolani, M. C. Neves, J. F. Botero, S. Latré, L. Zambenedetti, M. P. Barcellos, L. P. Gaspary, Optimal service function chain composition in network functions virtualization, in: Security of Networks and Services in an All-Connected World: 11th IFIP WG 6.6 International Conference on Autonomous Infrastructure, Management, and Security, AIMS 2017, Zurich, Switzerland, July 10-13, 2017, Proceedings, 2017, pp. 62–76.

[168] L. Ruiz, R. J. Barroso, I. De Miguel, N. Merayo, J. C. Aguado, R. De La Rosa, P. Fernández, R. M. Lorenzo, E. J. Abril, Genetic algorithm for holistic VNF-mapping and virtual topology design, IEEE Access 8 (2020) 55893–55904. doi:10.1109/ACCESS.2020.2982018.

[169] M. T. Beck, J. F. Botero, Coordinated Allocation of Service Function Chains, in: 2015 IEEE Global Communications Conference (GLOBECOM), 2015, pp. 1–6. doi:10.1109/GLOCOM.2015.7417401.

[170] J. Cao, Y. Zhang, W. An, X. Chen, J. Sun, Y. Han, VNF-FG design and VNF placement for 5G mobile networks, Science China Information Sciences 60 (2017) 1–15. doi:10.1007/S11432-016-9031-X/METRICS.

[171] S. Sahhaf, W. Tavernier, J. Czentye, B. Sonkoly, P. Sköldström, D. Jocha, J. Garay, Scalable Architecture for Service Function Chain Orchestration, in: European Workshop on Software Defined Networks, EWSDN, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 19–24. doi:10.1109/EWSDN.2015.55.

[172] Z. Wang, J. Zhang, T. Huang, Y. Liu, Service Function Chain Composition, Placement, and Assignment in Data Centers, IEEE Transactions on Network and Service Management 16 (2019) 1638–1650. doi:10.1109/TNSM.2019.2933872.

[173] S. Tavakoli-Someh, M. H. Rezvani, Multi-objective virtual network function placement using NSGA-II meta-heuristic approach, Journal of Supercomputing 75 (2019) 6451–6487. doi:10.1007/S11227-019-02849-Y.

[174] M. Jalalitabar, G. Luo, C. Kong, X. Cao, Service function graph design and mapping for NFV with priority dependence, in: IEEE Global Communications Conference, GLOBECOM, 2016, pp. 1–5. doi:10.1109/GLOCOM.2016.7842263.

[175] M. T. Beck, J. F. Botero, Scalable and coordinated allocation of service function chains, Computer Communications 102 (2016) 78–88. doi:10.1016/j.comcom.2016.09.010.

[176] J. Gil-Herrera, J. F. Botero, A scalable metaheuristic for service function chain composition, in: 2017 IEEE 9th Latin-American Conference on Communications, LATINCOM 2017, volume 2017-Janua, Institute of Electrical and Electronics Engineers Inc., 2017, pp. 1–6. doi:10.1109/LATINCOM.2017.8240194.

[177] H. Cao, Z. Qu, Y. Xue, L. Yang, Efficient virtual network embedding algorithm based on restrictive selection and optimization theory approach, China Communications 14 (2017) 39–60. doi:10.1109/CC.2017.8107631.

[178] F. Carpio, W. Bziuk, A. Jukan, Replication of Virtual Network Functions: Optimizing link utilization and resource costs, 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2017 - Proceedings (2017) 521–526. doi:10.23919/MIPRO.2017.7973481.

[179] M. T. Beck, J. F. Botero, K. Samelin, Resilient allocation of service Function chains, in: 2016 IEEE Conference on Network Function Virtualization and Software Defined Networks, NFV-SDN 2016, Institute of Electrical and Electronics Engineers Inc., 2016, pp. 128–133. doi:10.1109/NFV-SDN.2016.7919487.

[180] Y. T. Woldeyohannes, A. Mohammadkhan, K. K. Ramakrishnan, Y. Jiang, A scalable resource allocation scheme for NFV: Balancing utilization and path stretch, in: 21st Conference on Innovation in Clouds, Internet and Networks, ICIN 2018, Institute of Electrical and Electronics Engineers Inc., 2018, pp. 1–8. doi:10.1109/ICIN.2018.8401631.

[181] L. S. Globa, V. A. Prokopets, N. A. Gvozdetska, Prognostic-Reactive NFV Resource Allocation Method for Implementation in Virtualized Mobile Network EPC of Ukraine, in: 2018 IEEE International Black Sea Conference on Communications and Networking, BlackSeaCom 2018, IEEE, 2018, pp. 1–5. doi:10.1109/BlackSeaCom.2018.8433602.

[182] B. Spinnewyn, J. F. Botero, C. Donato, S. Latre, Effective NFV orchestration for wide-ranging services across heterogeneous cloud networks, 2019 IFIP/IEEE Symposium on Integrated Network and Service Management, IM 2019 (2019) 107–115.

[183] J. Chen, J. Chen, R. Hu, MoclusVNF: Leveraging multi-objective for scalable NFV orchestration, in: ACM International Conference Proceeding Series, Association for Computing Machinery, 2019, pp. 27–33. doi:10.1145/3361821.3361837.

[184] V. Fulber-Garcia, M. C. Luizelli, C. R. dos Santos, E. J. Spinosa, E. P. Duarte, Customizable Mapping of Virtualized Network Services in Multi-datacenter Environments Based on Genetic Metaheuristics, Journal of Network and Systems Management 31 (2023) 1–31. doi:10.1007/S10922-023-09760-1/FIGURES/18.

[185] M. Pozza, A. Patel, A. Rao, H. Flinck, S. Tarkoma, Composing 5g network slices by co-locating vnfs in μslices, in: 2019 IFIP Networking Conference, IFIP Networking 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–9. doi:10.23919/IFIPNETWORKING.2019.8816860.

[186] Y. Yue, B. Cheng, M. Wang, B. Li, X. Liu, J. Chen, Throughput Optimization and Delay Guarantee VNF Placement for Mapping SFC Requests in NFV-Enabled Networks, IEEE Transactions on Network and Service Management 18 (2021) 4247–4262. doi:10.1109/TNSM.2021.3087838.

[187] Y. Yue, B. Cheng, B. Li, M. Wang, X. Liu, Throughput optimization VNF placement for mapping SFC requests in MEC-NFV enabled networks, in: Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM, Association for Computing Machinery, 2020, pp. 920–922. doi:10.1145/3372224.3418169.

[188] B. Németh, J. Czentye, G. Vaszkun, L. Csikor, B. Sonkoly, Customizable real-time service graph mapping algorithm in carrier grade networks, in: 2015 IEEE Conference on Network Function Virtualization and Software

Defined Network, NFV-SDN 2015, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 28–30. doi:10.1109/NFV-SDN.2015.7387400.

[189] T. Lukovszki, S. Schmid, Online admission control and embedding of service chains, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9439 (2015) 104–118. doi:10.1007/978-3-319-25258-2{_}8/COVER.

[190] S. Xie, J. Ma, J. Zhao, FlexChain: Bridging Parallelism and Placement for Service Function Chains, IEEE Transactions on Network and Service Management 18 (2021) 195–208. doi:10.1109/TNSM.2020.3047834.

[191] T. W. Kuo, B. H. Liou, K. C. J. Lin, M. J. Tsai, Deploying chains of virtual network functions: On the relation between link and server usage, in: IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications, volume 2016-July, Institute of Electrical and Electronics Engineers Inc., 2016, pp. 1–9. doi:10.1109/INFOCOM.2016.7524565.

[192] W. Mao, L. Wang, J. Zhao, Y. Xu, Online Fault-tolerant VNF Chain Placement: A Deep Reinforcement Learning Approach, in: 2020 IFIP Networking Conference (Networking), 2020, pp. 163–171.

[193] S. B. Chetty, H. Ahmadi, A. Nag, Virtual Network Function Embedding under Nodal Outage using Reinforcement Learning, International Symposium on Advanced Networks and Telecommunication Systems, ANTS 2020-Decem (2020). doi:10.1109/ANTS50601.2020.9342803.

[194] S. B. Chetty, H. Ahmadi, M. Tornatore, A. Nag, Dynamic Decomposition of Service Function Chain Using a Deep Reinforcement Learning Approach, IEEE Access 10 (2022) 111254–111271. doi:10.1109/ACCESS.2022.3215744.

[195] L. Wang, W. Mao, J. Zhao, Y. Xu, DDQP: A Double Deep Q-Learning Approach to Online Fault-Tolerant SFC Placement, IEEE Transactions on Network and Service Management 18 (2021) 118–132. doi:10.1109/TNSM.2021.3049298.

[196] I. Jang, D. Suh, S. Pack, G. Dán, Joint optimization of service function placement and flow distribution for service function chaining, IEEE Journal on Selected Areas in Communications 35 (2017) 2532–2541. doi:10.1109/JSAC.2017.2760162.

[197] M. Laroui, H. I. Khedher, H. Moungla, H. Afifi, Service Function Chains multi-resource orchestration in Virtual Mobile Edge Computing, Computer Networks 224 (2023) 109582. doi:10.1016/J.COMNET.2023.109582.

[198] M. A. T. Nejad, S. Parsaeefard, M. A. Maddah-Ali, T. Mahmoodi, B. H. Khalaj, VSPACE: VNF Simultaneous Placement, Admission Control

and Embedding, IEEE Journal on Selected Areas in Communications 36 (2018) 542–557. doi:10.1109/JSAC.2018.2815318.

[199] S. M. Araújo, F. S. De Souza, G. R. Mateus, A Composition Selection Mechanism for Chaining and Placement of Virtual Network Functions, in: 15th International Conference on Network and Service Management, CNSM 2019, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–5. doi:10.23919/CNSM46954.2019.9012746.

[200] L. Slim, F. Bannour, Hierarchical Multi-Agent Deep Reinforcement Learning with an Attention-based Graph Matching Approach for Multi-Domain VNF-FG Embedding, in: IEEE Global Communications Conference, GLOBECOM, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 2105–2110. doi:10.1109/GLOBECOM54140.2023.10437970.

[201] Z. Liu, Z. Shu, S. Chen, Y. Zhong, J. Lin, Low-latency Virtual Network function Scheduling Algorithm Based on Deep Reinforcement Learning, Computer Networks 246 (2024) 110418. doi:10.1016/j.comnet.2024.110418.

[202] R. Sahraoui, O. Houidi, F. Bannour, Energy-Aware VNF-FG Placement with Transformer-based Deep Reinforcement Learning, in: NOMS 2024-2024 IEEE Network Operations and Management Symposium, 2024, pp. 1–9. doi:10.1109/NOMS59830.2024.10575040.

[203] S. B. Chetty, A. Nag, A. Al-Tahmeesschi, Q. Wang, B. Canberk, J. Marquez-Barja, H. Ahmadi, Optimized Resource Allocation for Cloud-Native 6G Networks: Zero-Touch ML Models in Microservices-Based VNF Deployments, IEEE Network 39 (2025) 235–243. doi:10.1109/MNET.2024.3486623.

[204] G. L. Santos, T. Lynn, J. Kelner, P. T. Endo, Availability-aware and energy-aware dynamic SFC placement using reinforcement learning, Journal of Supercomputing 77 (2021) 12711–12740. doi:10.1007/S11227-021-03784-7/TABLES/4.

[205] R. Solozabal, J. Ceberio, A. Sanchoyerto, L. Zabala, B. Blanco, F. Liberal, Virtual Network Function Placement Optimization with Deep Reinforcement Learning, IEEE Journal on Selected Areas in Communications 38 (2020) 292–303. doi:10.1109/JSAC.2019.2959183.

[206] A. Varasteh, B. Madiwalar, A. Van Bemten, W. Kellerer, C. Mas-Machuca, Holu: Power-Aware and Delay-Constrained VNF Placement and Chaining, IEEE Transactions on Network and Service Management 18 (2021) 1524–1539. doi:10.1109/TNSM.2021.3055693.

[207] B. Farkiani, B. Bakhshi, S. A. Mirhassani, A Fast Near-Optimal Approach for Energy-Aware SFC Deployment, IEEE Transactions on Network and Service Management 16 (2019) 1360–1373. doi:10.1109/TNSM.2019.2944023.

[208] S. Kim, Y. Han, S. Park, An Energy-Aware Service Function Chaining and Reconfiguration Algorithm in NFV, in: 2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems (FAS*W), 2016, pp. 54–59. doi:10.1109/FAS-W.2016.24.

[209] M. M. Tajiki, S. Salsano, L. Chiaraviglio, M. Shojafar, B. Akbari, Joint Energy Efficient and QoS-Aware Path Allocation and VNF Placement for Service Function Chaining, IEEE Transactions on Network and Service Management 16 (2019) 374–388. doi:10.1109/TNSM.2018.2873225.

[210] V. R. Chintapalli, S. B. Korrapati, B. R. Tamma, A. Franklin, NUMASFP: NUMA-Aware Dynamic Service Function Chain Placement in Multi-Core Servers, in: 2022 14th International Conference on COMmunication Systems and NETworkS, COMSNETS 2022, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 181–189. doi:10.1109/COMSNETS53615.2022.9668603.

[211] Y. Liu, H. Lu, X. Li, D. Zhao, An approach for service function chain reconfiguration in network function virtualization architectures, IEEE Access 7 (2019) 147224–147237. doi:10.1109/ACCESS.2019.2946648.

[212] T. Wang, Q. Fan, X. Li, X. Zhang, Q. Xiong, S. Fu, M. Gao, DRL-SFCP: Adaptive Service Function Chains Placement with Deep Reinforcement Learning, in: IEEE International Conference on Communications, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 1–6. doi:10.1109/ICC42927.2021.9500964.

[213] Y. T. Chen, W. Liao, Mobility-Aware Service Function Chaining in 5G Wireless Networks with Mobile Edge Computing, in: IEEE International Conference on Communications, volume 2019-May, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1–6. doi:10.1109/ICC.2019.8761306.

[214] M. Bagaa, T. Taleb, A. Ksentini, Service-aware network function placement for efficient traffic handling in carrier cloud, in: IEEE Wireless Communications and Networking Conference, WCNC, Institute of Electrical and Electronics Engineers Inc., 2014, pp. 2402–2407. doi:10.1109/WCNC.2014.6952725.

[215] Z. Xu, C. Jin, F. Yang, Z. Ding, S. Du, G. Zhao, Coordinated Resource Allocation with VNFs Precedence Constraints in Inter-Datacenter Networks over Elastic Optical Infrastructure, 2018 IEEE World Symposium on Communication Engineering, WSCE 2018 (2018) 1–6. doi:10.1109/WSCE.2018.8690540.

[216] P. K. Thiruvasagam, V. J. Kotagi, C. Siva Ram Murthy, A Reliability-Aware, Delay Guaranteed, and Resource Efficient Placement of Service Function Chains in Softwarized 5G Networks, IEEE Transactions on Cloud Computing 10 (2022) 1515–1531. doi:10.1109/TCC.2020.3020269.

[217] H. Moens, F. D. Turck, VNF-P: A model for efficient placement of virtualized network functions, in: 10th International Conference on Network and Service Management (CNSM) and Workshop, 2014, pp. 418–423. doi:10.1109/CNSM.2014.7014205.

[218] A. Pentelas, G. Papathanail, I. Fotoglou, P. Papadimitriou, Network Service Embedding with Multiple Resource Dimensions, Proceedings of IEEE/IFIP Network Operations and Management Symposium 2020: Management in the Age of Softwarization and Artificial Intelligence, NOMS 2020 (2020). doi:10.1109/NOMS47738.2020.9110440.

[219] J. Elias, F. Martignon, S. Paris, J. Wang, Efficient Orchestration Mechanisms for Congestion Mitigation in NFV: Models and Algorithms, IEEE Transactions on Services Computing 10 (2017) 534–546. doi:10.1109/TSC.2015.2498176.

[220] Z. Xu, Q. Han, B. Cheng, M. Niu, J. Chen, HASP: High Availability SFC Placement Approach in Data Center Network, in: 2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), 2021, pp. 448–455. doi:10.1109/HPCC-DSS-SmartCity-DependSys53884.2021.00083.

[221] P. Bellavista, F. Callegati, W. Cerroni, C. Contoli, A. Corradi, L. Foschini, A. Pernafini, G. Santandrea, Virtual network function embedding in real cloud environments, Computer Networks 93 (2015) 506–517. doi:10.1016/J.COMNET.2015.09.034.

[222] C. Ren, J. Zhang, Y. Wang, Y. Li, On Efficient VNF-FG Design in IoT Networks, IEEE Transactions on Network and Service Management 21 (2024) 4531–4542. doi:10.1109/TNSM.2024.3392976.

[223] Z. Zheng, J. Bi, H. Yu, H. Wang, C. Sun, H. Hu, J. Wu, Z. Zheng, J. Shen, C. Miao, C. Sun, H. Hu, J. Bi, J. Wu, J. Wang, Octans: Optimal Placement of Service Function Chains in Many-Core Systems, IEEE Transactions on Parallel and Distributed Systems 32 (2021) 2202–2215. doi:10.1109/TPDS.2021.3063613.

[224] M. Xia, M. Shirazipour, Y. Zhang, H. Green, A. Takacs, Network function placement for NFV chaining in packet/optical datacenters, Journal of Lightwave Technology 33 (2015) 1565–1570. doi:10.1109/JLT.2015.2388585.

[225] R. A. Addad, M. Bagaa, T. Taleb, D. L. C. Dutra, H. Flinck, Optimization model for cross-domain network slices in 5g networks, IEEE Transactions on Mobile Computing 19 (2020) 1156–1169. doi:10.1109/TMC.2019.2905599.

[226] P. Khosravian, S. Emadi, G. Mirjalily, B. Zamani, IETF-based Finite Automaton for Service Composition in Service Function Chaining, Wireless Personal Communications 114 (2020) 1235–1247. doi:10.1007/S11277-020-07417-9/FIGURES/6.

[227] Q. Zhang, X. Wang, I. Kim, P. Palacharla, T. Ikeuchi, Vertex-centric computation of service function chains in multi-domain networks, in: IEEE NETSOFT 2016 - 2016 IEEE NetSoft Conference and Workshops: Software-Defined Infrastructure for Networks, Clouds, IoT and Services, Institute of Electrical and Electronics Engineers Inc., 2016, pp. 211–218. doi:10.1109/NETSOFT.2016.7502415.

[228] S. Clayman, E. Maini, A. Galis, A. Manzalini, N. Mazzocca, The dynamic placement of virtual network functions, IEEE/IFIP NOMS 2014 - IEEE/IFIP Network Operations and Management Symposium: Management in a Software Defined World (2014). doi:10.1109/NOMS.2014.6838412.

[229] J. J. Alves Esteves, A. Boubendir, F. Guillemin, P. Sens, Heuristic for Edge-enabled Network Slicing Optimization using the "Power of Two Choices", in: 2020 16th International Conference on Network and Service Management (CNSM), 2020, pp. 1–9. doi:10.23919/CNSM50824.2020.9269099.

[230] R. Riggio, A. Bradai, D. Harutyunyan, T. Rasheed, T. Ahmed, Scheduling wireless virtual networks functions, IEEE Transactions on Network and Service Management 13 (2016) 240–252. doi:10.1109/TNSM.2016.2549563.

[231] D. Medhi, K. Ramasamy, Routing and Traffic Engineering in Data Center Networks, Network Routing (2018) 396–422. doi:10.1016/B978-0-12-800737-2.00014-4.

[232] D. L. Mills, H. W. Braun, The NSFNET backbone network, Proceedings of the ACM Workshop on Frontiers in Computer Communications Technology, SIGCOMM 1987 (1987) 191–196. doi:10.1145/55482.55502;JOURNAL:JOURNAL:ACMCONFERENCES;PAGEGROUP:STRING:PUBLICATION.

[233] J. I. Castillo-Velazquez, I. Munoz-Martinez, J. A. DIaz-Ramirez, E. F. Ordonez-Morales, Management emulation for GEANT advanced network: 2020 topology under IPv6, 2020 IEEE ANDESCON, ANDESCON 2020 (2020). doi:10.1109/ANDESCON50619.2020.9271972.

[234] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, S. Lu, BCube: A high performance, server-centric network architecture for modular data centers, Computer Communication Review 39 (2009) 63–74. doi:10.1145/1594977.1592577;WGROUP:STRING:ACM.

[235] J. I. Castillo-Velazquez, D. J. Serrano-Martinez, A. Morales, Emulation of the connectivity of backbone and management for the layer 3 service of INTERNET2: 2016 topology, 2017 IEEE 37th Central America and Panama Convention, CONCAPAN 2017 2018-January (2017) 1–4. doi:10.1109/CONCAPAN.2017.8278476.

[236] N. Spring, R. Mahajan, D. Wetherall, T. Anderson, Measuring ISP topologies with Rocketfuel, IEEE/ACM Transactions on Networking 12 (2004) 2–16. doi:10.1109/TNET.2003.822655.

[237] Z. Xu, L. Zhuang, W. Zhuang, Y. Hu, W. Mo, Z. Wang, Meta Relational Learning-Based Service-Tailored VNF Deployment for B5G Network Slice, IEEE Internet of Things Journal 10 (2023) 22542–22554. doi:10.1109/JIOT.2023.3303452.

[238] P. Harvey, A. Tatar, P. Imai, L. Wong, L. Bringuier, Evolutionary Autonomous Networks, Journal of ICT Standardization 9 (2021) 201–228. doi:10.13052/jicts2245-800X.927.

[239] H. M. A. Fahmy, Simulators and Emulators for WSNs, in: Concepts, Applications, Experimentation and Analysis of Wireless Sensor Networks, Springer Nature Switzerland, Cham, 2023, pp. 547–663. doi:10.1007/978-3-031-20709-9{_}8.

[240] M. Wang, B. Cheng, J. Chen, Poster:A Linear Programming Approach for SFC Placement in Mobile Edge Computing, in: Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM, volume 19, Association for Computing Machinery, 2019, pp. 1–3. doi:10.1145/3300061.3343394.