

JCST Papers

Only for Academic and Non-Commercial Use

Thanks for Reading!

[Survey](#)

[Computer Architecture and Systems](#)

[Artificial Intelligence and Pattern Recognition](#)

[Computer Graphics and Multimedia](#)

[Data Management and Data Mining](#)

[Software Systems](#)

[Computer Networks and Distributed Computing](#)

[Theory and Algorithms](#)

[Emerging Areas](#)



JCST WeChat

Subscription Account

JCST URL: <https://jct.ac.cn>

SPRINGER URL: <https://www.springer.com/journal/11390>

E-mail: jct@ict.ac.cn

Online Submission: <https://mc03.manuscriptcentral.com/jct>

Twitter: JCST_Journal

LinkedIn: Journal of Computer Science and Technology

Virtualization, Cloudification, and Service Orientation of Network: A Systematic Review

Shuo Quan¹ (全 硕), *Member, CCF, ACM*, Shen Gao¹ (高 深), *Member, CCF*
Jian-Song Zhang¹ (张健松), *Member, CCF*, and Jie Wu^{1,2}, *Fellow, CCF, IEEE*

¹ *Cloud Computing Research Institute, China Telecom, Beijing 100191, China*

² *Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122, U.S.A.*

E-mail: quansh@chinatelecom.cn; gaos13@chinatelecom.cn; zhangjs15@chinatelecom.cn; wujie@chinatelecom.cn

Received September 11, 2024; accepted December 4, 2024.

Abstract Innovations in new applications and technological advancements are driving the evolution of network architectures towards flexibility and automation. Network Function Virtualization (NFV) deploys Network Functions (NFs) as software applications onto cloud infrastructures, redefining the development, deployment, and operation models of communication networks, thereby meeting the evolution demands of networks. However, after more than a decade of development, the progress of network service operators in NFV has not met expectations, partly because some key technologies remain unresolved. To accelerate the large-scale commercial use of NFV, this paper focuses on reviewing relevant literature from the past five years. Based on practical applications and insights into future trends, we explore the three directions of network virtualization, network cloudification, and network service orientation. We investigate the most representative technologies and the latest research progress in these fields, analyze the current problems and challenges, and provide corresponding suggestions on how to deal with them. Finally, we forecast future directions of technological development.

Keywords network function virtualization, network virtualization, network cloudification, network service orientation

1 Introduction

Telecommunication (telecom) networks provide networking backbone for the Internet, which now also connect various datacenters to billions of user devices to provide cloud computing. Traditional telecom networks were built with dedicated networking devices. These devices were supplied by different vendors and they can interoperate with each other by following the same pre-defined networking protocols. However, this traditional way for telecom network construction has become sluggish, inflexible, and inefficient. Especially in the recent decade, the Internet traffic volume has been growing exponentially, and applications have been more and more diverse. We have witnessed a lot of changes, for example, multimedia has become the dominating traffic type, big da-

ta and artificial intelligence are ubiquitous, new applications such as autonomous driving and metaverse have become important, networking for emerging scenarios such as AIoT (AI IoT), industry automation and aircraft in low-altitude are necessarily considered, etc.

Exploring new means for telecom network construction has been an industry's joint effort during the recent decade. In 2012, the European Telecommunications Standards Institute (ETSI) introduced the concept of Network Function Virtualization (NFV)^①, and set up the first industry specification group for NFV. The main idea of NFV is to bring those successful experiences in Information Technology (IT) and Computer Science (CS) into Communications Technology (CT) industry, thereby building an evolved architecture for telecom networks, which is

Survey

^①<https://www.etsi.org/technologies/nfv>, May 2025.

©Institute of Computing Technology, Chinese Academy of Sciences 2025

scalable, flexible, reliable, and efficient. Afterwards, NFV has carried out a series of work, from replacing dedicated hardware with Commercial Off-The-Shelf (COTS) servers and introducing Management and Orchestration (MANO), to also integrating with cloud-native technologies and providing support for multi-cloud and hybrid cloud, and further embracing Service-Based Architecture (SBA)^②. Till today, the telecom network architecture has fully evolved, which now can be divided into three layers, i.e., virtualization, cloudification, and service orientation.

Taking the 5G private network as an example, in response to the different network needs of various industries, China Telecom offers three models catering to customers with wide-area priority, latency sensitivity, and security sensitivity. Among them, the “BiLin” model, as shown in Fig.1, deploys a 5G data plane User Plane Function (UPF) in the edge cloud close to the industrial park, achieving lower transmission latency, and deploys a 5G control plane in the central cloud, ensuring high reliability and stable, elastic operation. Network virtualization enables the deployment of NFs on COTS servers, reducing costs and improving scalability, and further allows deployment in the cloud. Network cloudification leverages the advantage of on-demand resource allocation, meeting the stable and elastic operational requirements of 5G Core (5GC) and reducing operational

complexity. Network service orientation enables the on-demand combination of different network function modules, providing customized network capabilities for customers, and reducing service activation time from months to minutes. The resulting low cost, easy maintenance, elasticity, flexibility, and business agility have become the main driving forces for the network’s evolution towards virtualization, cloudification, and service orientation.

Network Virtualization. It is similar to the virtualization in cloud computing. Virtualization techniques provide abstractions for computing, storage and networking resources, and isolation between software programs. Based on virtualization, multiple software programs can run independently on the same hardware. Network virtualization provides a new way for telecom network construction that COTS servers plus virtual machine or container software can be used to replace dedicated networking devices. More importantly, software can be decoupled from hardware, so that NFs can be implemented in software which is provided by different vendors from device suppliers. Moreover, Software-Defined Networking (SDN) can be implemented to control devices from multiple vendors. The most important technical issues include heterogeneous computing, high-performance virtualization, virtual deployment units, and end-to-end acceleration. We will elaborate these issues in Section 2.

Network Cloudification. It brings resource management and software deployment techniques from cloud computing into telecom network construction, for example, 1) using Cloud Infrastructure as a Service (IaaS) services to manage and orchestrate virtual machines, containers, and the virtual NFs (VNFs) running on them; 2) using Platform as a Service (PaaS) services to provide capabilities such as databases and middleware, which lowers the barrier to development; 3) leveraging Development and Operations (DevOps) and other automation tools to simplify operations and management tasks; 4) taking advantages of the cloud’s elasticity to break through the limitations of physical resources; 5) and capabilities like multiple Availability Zones (Multi-AZ) also provide high reliability and high availability for VNFs. Thus, the constantly changing business needs are met, and flexible network services are provided. The most important technical issues in the cloudification layer include cloud native, cloud pass, Reliability, Avail-

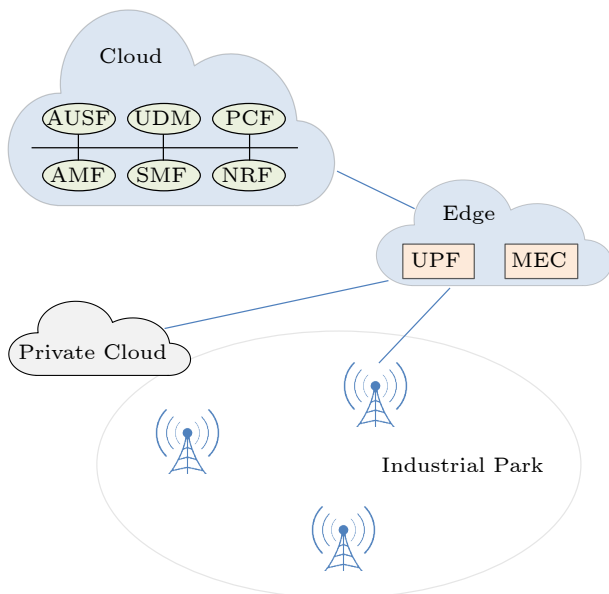


Fig.1. 5G network function deployment diagram.

^②European Telecommunications Standards Institute, Jun 2022. <https://docbox.etsi.org/ISG/NFV/Open/Other/ReleaseDocumentation/>, Mar. 2025.

ability, and Serviceability (RAS) capability, resiliency, scalability, and multi-cloud and distributed cloud. We will elaborate these issues in [Section 3](#).

Network Service Orientation. It transforms networking capabilities into services. To provide more elastic, scalable, flexible, customizable, and rapidly deployable network services, it is necessary to restructure NFs at the software architecture level. Based on the service-oriented architecture reconstruction, NFs are decomposed into smaller modules and encapsulated into atomic capabilities, allowing network services to be quickly developed based on these atomic capabilities. The most important technical issues include microservice architecture, service automation, and declarative API. We will elaborate these issues in [Section 4](#).

The core value of network virtualization lies in achieving the decoupling of software and hardware. And by leveraging the abstraction and isolation of resources, network virtualization masks the differences in underlying hardware for virtual NFs, which lays the foundation for the flexibility and scalability of networks. From the perspective of resource management, virtualization typically focuses on individual servers (ignoring the “many-to-one” scenario) and cannot provide management and scheduling across multiple clusters or even data centers. This requires the use of cloudification technology to achieve ultimate elasticity, high reliability, and automation capabilities on top of server clusters. On the other hand, network virtualization is also a prerequisite for cloudi-

fication, because without the standardized virtual environment provided by virtualization, cloudification would be impossible as it builds upon network virtualization to offer cloud capabilities. Network service orientation starts from the application level. By endowing applications with more flexible internal software architectures, it can fully leverage the advantages of the elastic, highly reliable infrastructures built by network virtualization and cloudification. In this way, it can provide network services that are elastic, customizable, and capable of rapid delivery. This relies on the elastic, highly reliable infrastructures built by network virtualization and network cloudification. Thus, their relationship is depicted in [Fig.2](#), with network virtualization at the bottom layer, network cloudification in the middle layer, and network service orientation at the top layer.

In this survey paper, we provide a systematic review for an evolved telecom network architecture by summarizing related work in literature. Meanwhile, we add our observations and understandings about major technical challenges, current progresses, and future directions. This is the first comprehensive survey for this topic. In [Section 6](#), we will also compare our work with related survey work.

After our extensive study, the major observation is that, although the three-layer architecture of virtualization, cloudification, and service orientation has been widely accepted, the technical challenges in each layer have not been fully addressed yet. In [Fig.3](#), we summarize main challenges and major technical top-

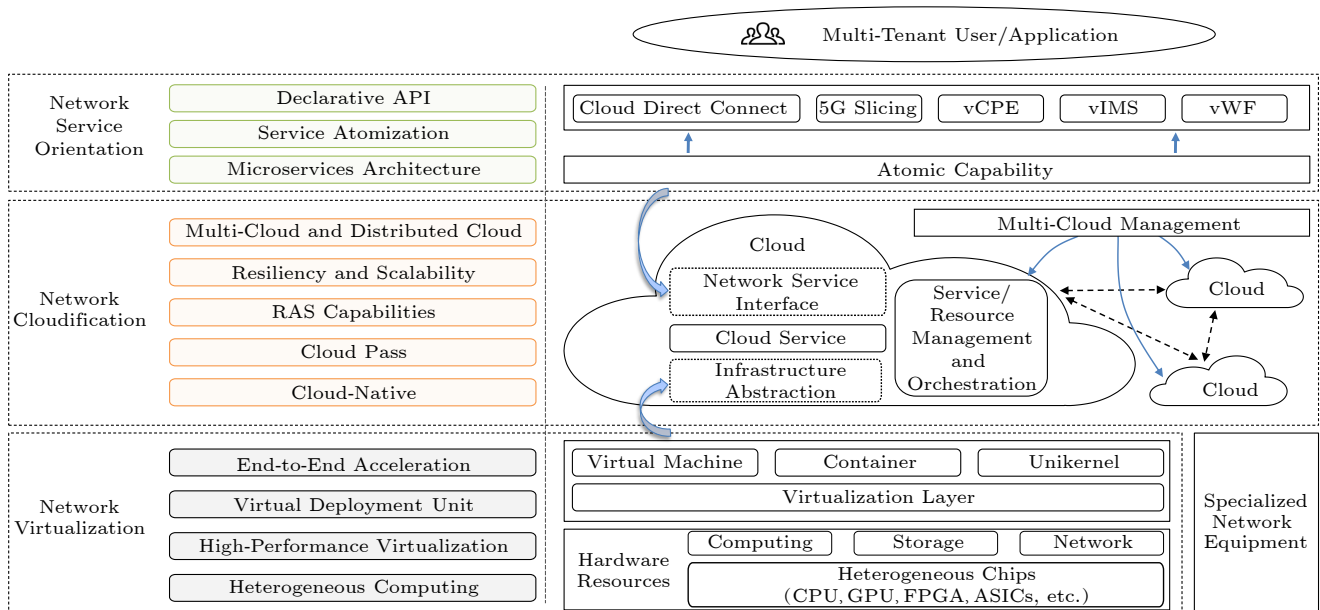


Fig. 2. Evolved telecom network architecture: virtualization, cloudification, and service orientation.

Layer	Main Challenges	Current Technologies	Future Directions
Network Service Orientation	The evolution of network services from closed to flexible assembly and open sharing. 1) How to construct an effective service model that progresses towards cloud-native and microservices. 2) The real-time, consistency, and effectiveness of large-scale network service configuration changes.	Microservices-Based Network Function Cloud-Native Platform Support	Cloud Network Convergence Declarative Intent-Driven
Network Cloudification	Networks have higher demands in terms of performance, reliability, operations, and maintenance. 1) The increasing demand for customization raises the threshold. How to use universal methods to meet these characteristics during the introduction of advanced cloud technologies. 2) How to collaborate with the cloud to fully leverage its capabilities.	Resiliency and Scalability Telco Cloud Platform Cross Layer Operation Reliability	Autonomous Network Multi-Cloud and Distributed Cloud Management
Network Virtualization	The rapid growth of network traffic increasingly highlights the challenges of virtualization performance. 1) Currently, CPU processing is still the main method, with less use of hardware acceleration, leading to performance bottlenecks. 2) There is a lack of systematic organization of acceleration solutions, and the implementation of end-to-end overall acceleration is needed.	Hardware Offloading Kernel-Based Network Protocol Bypass Virtualization Performance Optimization SFC Deployment	Heterogeneous Infrastructure Customizable Computing

Fig.3. Organization of technical contents (from Section 2 to Section 5) in this paper. SFC: service function chain.

ics for every layer. We also point out six future directions, each of which may involve one, two, or all three levels. The remaining technical content of this paper is therefore organized as follows. Sections 2, 3, and 4 each correspond to a level, discussing the challenges, major technical topics, and the author's observations for that layer. Section 5 explains the six future directions in details. Section 6 discusses related standards in industry and related surveys in literature. Finally, Section 7 concludes this paper.

In this paper, we make the following contributions.

- We provide an architectural view for evolved telecom networks which can be divided into three layers of network virtualization, network cloudification, and network service orientation. These layers are the main development directions and trends of NFV and are all crucial for large-scale commercial use.

- We summarize virtualization acceleration technologies and propose a classification method for acceleration technologies based on the deployment architecture of VNFs. This method allows for the combination of different acceleration technologies into a holistic solution in practical applications, tailored to the characteristics of VNFs, to provide end-to-end optimal acceleration capabilities.

- We describe the transition of NFs from virtualization to cloudification, and point out the challenges and opportunities presented by converging network services with cloud infrastructures. The focus is on creating multi-functional platforms that support dynamic management and provisioning of services over shared resources, enhancing performance, scalability,

and reliability, while tackling integration issues between cloud and network technologies.

- We analyze network service orientation from two aspects: the network function architecture and the infrastructure support, illustrating the considerations for evolving NFs towards a microservices architecture and the enhanced capabilities required by the cloud-native infrastructures to support these specialized applications.

- We explore the direction of future technological evolution, propose new solutions to the challenges and problems currently faced, and at the same time, envision the empowerment of network function virtualization, cloudification, and service orientation by integrating technologies such as cloud computing and AI.

2 Network Virtualization

Network virtualization is a foundational technology for NFs to migrate from specialized network hardware to general-purpose COTS servers, thereby reducing the dependency on expensive specialized hardware, supporting faster service deployment, and achieving higher resource utilization. COTS servers, because of their general-purpose design, have a considerable performance gap when compared with specialized network hardware. The primary objective of network virtualization is to achieve high performance on COTS servers.

2.1 Main Challenges

However, network virtualization brings perfor-

mance challenges which need to be addressed in the deployment of software-based NFs in practice.

- Firstly, CPU performance is usually the bottleneck. With the widespread application of technologies such as big data, the Internet of Things, large-scale content distribution networks, and 5G, network traffic has grown rapidly. Tsareva *et al.*^[1] analyzed a large backbone link in Japan and found that from 2007 to 2019, the total traffic of IPv4 and IPv6 increased more than tenfold. It is difficult to achieve high performance comparable to dedicated hardware-based NFs when running software NFs in a virtualized environment. For example, the processing latency of a fully virtualized gateway is up to 132 μ s, which is almost eight times the processing latency of a decomposed gateway (15 μ s)^[2]. If the throughput and performance equivalent to those of hardware-based NFs cannot be provided, applications may be hindered^[3].

- Secondly, hardware accelerators are still premature. To enhance the performance of COTS servers in processing network traffic, an increasing number of hardware accelerators are used to offload network traffic and increase the computing power of COTS servers. However, due to certain Research and Development barriers and the need to master technologies such as hardware-software integration and heterogeneous computing, it is usually difficult to apply them in practice. In the commercialization process, CPU processing is still the main focus. Further research and analysis are needed on the characteristics of different hardware acceleration devices, providing a heterogeneous computing framework to shield the differences in underlying hardware facilities, so as to break through performance bottlenecks.

- Finally, end-to-end optimization is complicated. The deployment architecture of VNFs is shown in Fig.4. Traffic first enters from a physical network interface card (NIC), and is received by an NIC driver, and then passed to a kernel network protocol stack. If an NF runs in a virtualized environment, it needs to be forwarded by a virtual switch to the runtime of a virtual machine, container, or unikernel, and finally forwarded to the NF. After NFs process the user's traffic, the traffic is returned along an original path. The performance of network virtualization is reflected in each link of the above chain, just like the barrel theory, where the slowest processing link determines the performance ceiling of the VNF, and congestion can also slow down the processing capability of the

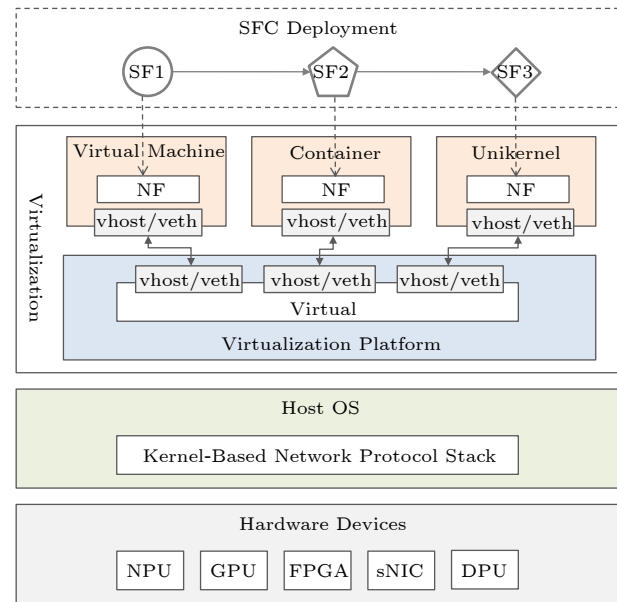


Fig.4. Network virtualization deployment architecture diagram.

entire chain. There is a lack of systematic organization of acceleration schemes in the existing literature to achieve end-to-end acceleration of the entire chain.

Based on the main challenges described above, we discuss four techniques in the following subsections. These techniques can be applied along the traffic path shown in Fig.4.

- *Hardware Offloading*: increasing processing speed by offloading tasks to hardware accelerators;
- *Kernel Bypass*: removing system overhead by a bypassing protocol stack in an operating system kernel;
- *Virtualization Performance Optimization*: reducing performance overhead caused by virtualization software;
- *Service Function Chain (SFC)*: achieving parallel acceleration, customized acceleration, etc., through orchestration of applications.

2.2 Hardware Offloading

Hardware offloading refers to the process of transferring certain tasks or computations from a computer's main processor to specialized hardware components (such as NIC or Graphics Processing Unit (GPU)). Hardware-specific devices can provide additional computing power on the basis of COTS servers, often offering higher processing efficiency due to their customized design. Different hardware exhibits distinct advantages in various scenarios, as can be seen in Table 1. Hardware offloading solutions are also one

Table 1. Hardware Acceleration Comparison

Hardware	Description	Effectiveness	Limitation
GPU	It has a large amount of parallel computing power, making it suitable for parallel computing	Improving the efficiency of network applications through multi-core parallelism and high memory bandwidth	High power consumption, relatively expensive, and low efficiency in non-parallel tasks
FPGA	Programmable hardware that can be configured with logic gates and storage units according to the requirements to perform specific computational tasks	High performance and dynamic instantiation of heterogeneous NFs	High development difficulty, and slightly inferior performance compared with specialized hardware
sNIC	Integrating processors and memories to achieve network programmability and task offloading	Implementing complex network data plane functions based on server architecture	Performance is limited by CPU processing capabilities
DPU	New type of programmable multi-core processor capable of performing high-speed data parsing and processing	Equipped with high-performance network interfaces for network protocol processing, switch routing computation, encryption, decryption, etc	The ecosystem is not mature enough, and there are compatibility issues with existing software and hardware ecosystems

of the main directions for building high-performance virtualized network applications.

GPU. The use of GPUs for processing network packets dates back to as early as 2010^[4]. By leveraging the high concurrency of GPUs for batch processing of packets, higher throughput can be achieved. Early research directions included how to maximize the use of hardware parallelism^[5], and optimize packet scheduling mechanisms through programming frameworks^[6], and applications in specific scenarios (intrusion detection, cryptography, etc.). In the latest research, to optimize the data transfer efficiency between a GPU and an NIC, Jung *et al.*^[7] utilized Single Root I/O Virtualization (SRIOV) to implement sharing between virtual machines and the host, while also passing the GPU through to the virtual machines. By using peer-to-peer-Direct Memory Access (DMA), packets from an NIC are directly mapped to a data buffer in a GPU memory pool, allowing the GPU to process its network traffic directly, without going through a host memory, which improves the performance by 2.3 times compared with a Data Plane Development Kit (DPDK) solution. Similarly, Kundel *et al.*^[8] used the peer-to-peer function of Peripheral Component Interconnect express (PCIe) to bypass a host, enabling direct connection between a GPU and an NIC, avoiding interference from an operating system and control applications, and achieving performance of over 10 Gbit/s. In addition, Guo *et al.*^[9], from the perspective of service chains, aimed to eliminate redundant data transmission between NFs, and proposed a data-sharing GPU-accelerated NFV system called Gaviss, which improved the performance by 2.5 times compared with the most advanced CPU-based methods.

Field-Programmable Gate Array (FPGA). Compared with the GPU solutions, FPGAs have the ad-

vantage of improving performance and reducing energy consumption. One of the challenges of FPGAs is programming with hardware description languages, which is often more difficult in terms of building logic and debugging codes. In an NFV scenario, P4^[10] and Click are considered to be two good choices^[3], where P4 can easily match a match-action network model, and Click has a rich set of development libraries and legacy codes. Similar to GPU research, optimizing transfer efficiency is also a direction of wisdom. Arikawa *et al.*^[11] reduced the memory copy overhead between an FPGA and a CPU through a high-bandwidth memory copy technology based on PCIe, achieving an improvement of about 14%. The latest research results show that using FPGAs can provide high-bandwidth transmission performance of over 90 Gbps for the entire network system^[12].

Smart NIC (sNIC). Smart network cards, compared with traditional ones, can take on more computational tasks and are programmable. The current hot research direction is to offload some traffic to the sNIC for processing based on extended Berkeley Packet Filter (eBPF). Lettieri *et al.*^[13] used the sNIC as an accelerator for stream processing operators and implemented offloading in eBPF, which improved performance by 1.6 times compared with a purely software approach. Abranches *et al.*^[14] used eXpress Data Path (XDP)/eBPF to offload network telemetry tasks to the smart network card to reduce the resource occupation of software network analysis and avoid unnecessary and redundant computations. sNIC has a good application prospect in edge scenarios, and is capable of solving issues such as insufficient computing power. Miano *et al.*^[15] built an efficient processing pipeline through an sNIC, using XDP/eBPF to complete part of the packet processing in the kernel space and a part in the sNIC, achieving a balance between traffic

sampling and aggregation in the kernel and traffic filtering in the hardware, thereby enhancing the processing capability of edge servers.

Data Processing Unit (DPU). Although sNICs can effectively offload data plane, they do not have a general-purpose processor like CPU, thus still requiring host CPUs for control plane management. As network speeds continue to increase, it will still consume a significant amount of valuable CPU resources. A DPU is a new type of processor that adds a CPU and a rich set of hardware acceleration units to the sNIC. NVIDIA's latest DPU product, BlueField-3, offers up to 400 Gbps network connectivity^[16], and it has a broad application prospect in data centers. Barsellotti *et al.*^[17] explored the application of DPU in edge scenarios, including infrastructure monitoring, low-energy cloud-network connections, and network security functions. Zhang *et al.*^[18] mentioned in the key directions of cloud-network integration the formation of a new type of data center network based on DPU, as depicted in Fig.5, breaking the boundaries between computing devices, storage devices, and network devices, reconstructing the infrastructure layer, and further unleashing the computational power efficiency.

2.3 Kernel-Based Network Protocol Bypass

Modern COTS servers are typically equipped with

multiple CPU processors, and their processing performance is gradually increasing. However, a core reason why they still cannot guarantee high performance in NFV applications is that an operating system's network protocol stack is not designed for rapid traffic processing like a network appliance, but rather for general communication^[19]. Therefore, many kernel bypass technologies have been introduced to improve performance by bypassing a kernel network protocol stack, such as DPDK, XDP, netmap, VPP, and Snabb. The kernel bypass technology allows applications to directly access and control hardware resources, reducing system interrupts, data copying, context switching, system calls, etc., thereby significantly improving network performance. DPDK and XDP/eBPF are two main representative technologies.

DPDK. DPDK bypasses the operating system kernel, moving most of packet processing to the user space, reducing the overhead and latency of the kernel layer, and can improve the packet processing performance by up to 10 times, achieving a throughput of over 80 million packets per second^③. DPDK has become relatively mature and is the most widely used kernel network protocol stack bypass technology. In recent years, there were less new research, mainly focusing on its use in new software and hardware environments. Although DPDK improves processing performance, the CPU overhead is not reduced. It uses

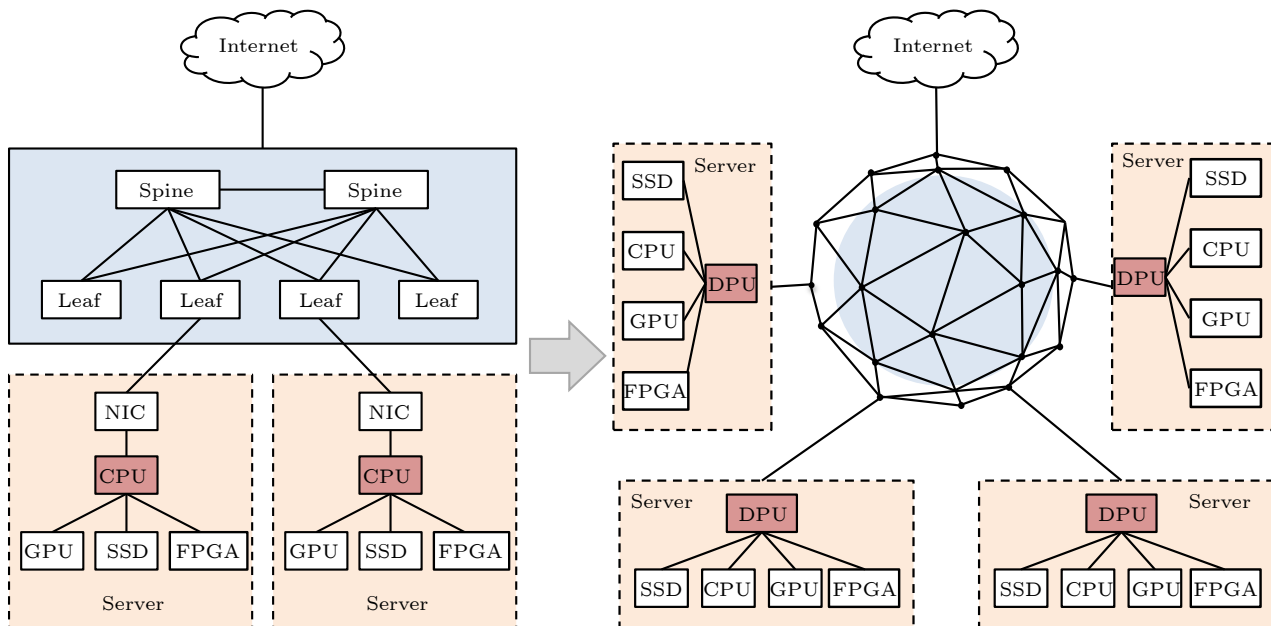


Fig.5. Based on DPU networking for new-type cloud computing illustration^[18].

^③Intel. Building enterprise-level cloud solutions with outscale. 2024. <https://www.intel.com/content/dam/www/public/us/en/documents/case-studies/xeon-e5-2660-family-ssd-s3700-series-dpdk-case-study.pdf>, Jun. 2024.

poll mode drivers (PMDs) to continuously scan NIC ports to detect if packets have arrived, which can waste CPU cycles. Wu *et al.*^[20] proposed a model for finding an optimal thread sleep time to reduce CPU usage and power consumption. Belkhiri *et al.*^[21] found that kernel bypass makes traditional network monitoring and debugging tools obsolete and proposed a performance analysis framework specifically for DPDK, allowing users to gain insights into the internal operation of the application and diagnose potential errors. Ayub *et al.*^[22] studied how to efficiently use DPDK in Kubernetes.

XDP/eBPF. XDP offers an alternative approach to bypass the kernel network protocol stack, as seen with DPDK. It allows network packets to run user-defined programs immediately upon reaching a network device interface, without waiting to be processed in the network protocol stack, thus achieving low latency and high throughput. eBPF can flexibly extend the functionality of XDP. Parola *et al.*^[23] believe that the high performance provided by technologies such as DPDK comes at the cost of strict resource partitioning, which is not suitable for edge data centers. Therefore, they use the XDP to process packets within the kernel, achieving a high level of integration with non-data plane applications. In the era of microservices, network function frameworks based on eBPF have attracted more attention. Miano *et al.*^[24] proposed a Polycube framework, which uses eBPF to handle NFs, providing an independent data plane and a control plane for each network function, achieving a great deal of flexible customization. The eBPF-based container network interface (CNI) plugin Cilium has also shown great potential. XDP/eBPF is often used to offload network traffic to hardware accelerators, which is discussed in [Subsection 2.2](#).

Comparison Between DPDK and XDP. We analyze DPDK and XDP in terms of technical maturity, application scenarios, and performance. AF_XDP was introduced in Linux kernel version 4.18 and still has significant room for optimization compared with DPDK. DPDK and AF_XDP have fundamental differences in principle. DPDK's kernel bypass strategy uses PMDs, while AF_XDP can be event-driven or polling, and its performance is lower than that of DPDK. Mostafa *et al.*^[25] also confirmed that an AF_XDP kernel driver is not so mature as a DPDK's user-space driver and has a lower processing speed than DPDK, which is also mentioned in [\[23\]](#). Regarding the use of DPDK and XDP, Zeng *et al.*^[26] made

full use of the characteristics of traffic and proposed another approach: using DPDK for L2/L3 traffic to achieve the "bump-in-the-wire" processing performance required by NFs, and using eBPF and a shared memory for L4/L7 traffic to achieve high-performance communication for NFs.

2.4 Virtualization Performance Optimization

Virtualization is a resource management technique that allows a single physical computer to run multiple operating systems, each of which can independently run different applications. Its core idea is to achieve resource sharing among multiple virtual machines or containers through abstraction. Although the virtualization technology can effectively improve the utilization of hardware, achieve flexibility and scalability, and simplify management and deployment, it also brings issues such as performance overhead and security isolation, which are particularly important in high-performance scenarios. How to enhance virtualization performance is a significant subject. Here, we analyze the optimization work of carrying NFs in the virtualized environment from three dimensions: virtualization platform, virtual switch, and virtual device interface.

Virtualization Platform. Currently widely used virtualization platforms include hypervisors and container engines. A virtual machine (VM) is a runtime environment created by simulating hardware resources on a physical server based on a hypervisor, featuring an independent operating system and library files, etc., providing kernel-level isolation capabilities with higher security performance. Because it includes a complete operating system, it consumes more resources and involves processes such as sensitive instruction switching between root and non-root modes, memory copying, and interrupt handling, leading to significant performance losses. To reduce these losses, technologies such as virtio and SR-IOV have been proposed. A container is a runtime environment built within an operating system based on a container engine; containers share the host's operating system kernel and, compared with virtual machines, have higher resource utilization and better performance but pose certain security risks^[27, 28].

Software tuning within virtualization platforms has been proven to achieve better results. Tran *et al.*^[29] tested the impact of CPU isolation and NUMA alignment on NF performance, confirming that these

optimization methods can effectively enhance network performance. In the scheme presented in [30], CPU pinning showed potential in optimizing the performance of container-based NFs. For memory-intensive NFs, using large pages can cache more content at once, effectively reducing the time spent on table lookups and thus improving performance. These configurations have achieved good results in industrial practice and require virtualization platforms to be set up when creating runtime environments.

We further discuss the advantages and disadvantages of runtime environments created based on different virtualization platforms as a reference for selecting runtime environments. The most commonly used are still virtual machines and containers. Tambe *et al.*[31], based on the MEC platform, studied the startup time, instantiation time, and service response time of virtual NFs on virtual machines and containers. They found that the performance of containers is superior to that of virtual machines, which is consistent with the conclusions in [28]. Ramanathan *et al.*[32] compared the performance of virtual machines and containers carrying EPC components in real-time migration and found that the real-time migration time and service interruption time of containers were reduced by about 70%. In addition, Ventre *et al.*[33] introduced the unikernel to optimize the virtualization resource instantiation processes, reducing the instantiation time of micro VNFs to less than 1 second, but did not deeply discuss the handling of network performance. New runtime solutions such as lightweight virtual machines and secure containers have not yet been used in network virtualization scenarios.

There are also virtualization platforms that have been customized for NFV scenarios, but these modifications are not suitable for general applications. Considering the evolution of networks towards cloudification and service orientation, non-general technical solutions may encounter obstacles in practice and are not discussed here.

Virtual Switch (vSwitch). vSwitch manages and forwards traffic in a virtualized environment, and can be divided into two implementation methods: kernel-based and kernel-bypass. Considering performance factors, the kernel-bypass technology is usually used.

We discussed the application of DPDK and XDP on vSwitch in Subsection 2.3. Taking the typical implementation of vSwitch and Open vSwitch (OVS), as an example, an OVS's architecture is divided into user space processes and kernel modules. By using DPDK to move all OVS components to user space, the OVS-DPDK solution^④ is widely used. Yang^⑤ believed that there are compatibility issues with using DPDK to accelerate OVS, including configuration management tools and CPU pinning, which are rejected in public cloud deployments for NFV customers due to cost and management issues^⑥. Therefore, Tu *et al.*[34] preferred to use the latest Linux network development AX_XDP technology and proposed the OVS-AF-XDP solution to achieve performance comparable to OVS-DPDK, while also achieving software maintainability and system operability.

Virtual Device Interface. Common virtualized device interfaces include veth pairs, TUN/TAP, and virtio/vhost. The veth device is used to reverse the direction of communication data. The data to be sent is converted into data that needs to be received and reprocessed by the kernel network layer, thereby indirectly completing data injection. Veth usually appears in pairs and is used for container networking. TUN/TAP is a device that connects user-space programs with kernel protocol stack and is commonly used to connect virtual machines and containers to vSwitch. Virtio and vhost are two types of devices in semi-virtualized solutions. Virtio is the front-end device that receives internal requests from virtual machines and containers, while vhost is the back-end device that processes requests from virtio.

2.5 SFC Deployment

Network services are typically composed of multiple NFs in SFCs or a network function chain (NFC). High performance in virtual networks can be achieved not only by optimizing individual NFs but also by considering the service chain as a whole, including parallel acceleration, customized acceleration, and other methods.

Parallel acceleration involves changing the sequential processing of packets in the service chain to paral-

^④R. Giller. Open vSwitch with DPDK overview. 2022. <https://www.intel.com/content/www/us/en/developer/articles/technical/open-vswitch-with-dpdk-overview.html>, Mar. 2025.

^⑤Yi Yang. Ovs dpdk issues in openstack and kubernetes and solutions. 2019.

^⑥VMWare. Update to VMWare per-CPU pricing model. 2020. <https://www.vmware.com/company/news/updates/cpu-pricing-modelupdate-feb-2020.html>, Jun. 2024.

lel processing, reducing network latency and improving performance. A challenge here is to ensure that the actions of NFs on packets do not conflict with each other and do not lead to additional resource waste. The study in [35] demonstrated the feasibility and proposed a high-performance framework NFP. It can automatically compile into a high-performance parallel graph for deployment based on the topology and dependencies of NFs, and supports parallelism with mechanisms such as packet replication, distributed transmission, and packet merging. In recent work, [36] studied the joint optimization of SFC parallelism and layout to achieve a higher acceptance rate. Zheng *et al.*[37] proposed a logarithmic approximation algorithm to jointly optimize packet processing and propagation latency. In addition to focusing on performance improvement, Agarwal *et al.*[38] addressed the issue of resource waste in packet replication/merging and packet deposition from the perspective of resource consumption. As technology matures, Chintapalli *et al.*[39] began to pay attention to the energy consumption issues of parallel acceleration.

When deploying NFs, it is necessary to combine the characteristics of the application to make better acceleration decisions. Zhang *et al.*[40] proposed the Flurries platform according to the needs of network function diversity, fine-grained customization of service chains, and flow-level performance guarantees, to accelerate a large number of lightweight NFs with short life cycles. Zeng *et al.*[41] proposed a framework called Gost that minimizes the task scheduling of end-to-end latency for multiple different NFCs. The paper utilizes the narrow application characteristics of VNFs, invents a pipeline based on device overlap and thread-level GPU resource allocation and kernel management strategies, and proposes the LSSTG algorithm to find a suboptimal solution.

2.6 Observations and Guidelines for Practical Use

Hardware acceleration introduces additional computing power to enhance the processing limits of COTS servers. Kernel bypass and virtualization performance optimization reduces overhead and helps to approach the processing limits of COTS servers. Hardware acceleration can achieve better performance but is usually limited to specific workloads and also brings additional costs. Software acceleration methods are easier to deploy but can be limited by hardware and may not meet the requirements of high performance

and high bandwidth. We point out three guidelines that can be applied in practice.

1) *Different Acceleration Schemes Have Their Respective Applicable Scenarios.* Taking east-west traffic and north-south traffic as examples, as shown in Fig.6, we compare the acceleration effects of these two scenarios under SR-IOV and OVS-DPDK. In the east-west traffic scenario, data does not leave a physical network card, and the OVS-DPDK solution achieves data exchange between VNFs directly through memory copying, which has higher performance than SR-IOV, which completes data exchange through the physical network card. In the north-south traffic scenario, data leaves the physical network card, and the SR-IOV solution bypasses the transmission in the virtualization layer through direct pass-through, offering better performance than OVS-DPDK.

2) *Different Applications Exhibit Variability in Acceleration Effects.* A single VNF has submodules with different computational, storage, and communication requirements. An efficient solution needs to recognize the implementation effects of VNFs on different devices to better utilize heterogeneous NFVI capabilities. At the same time, it should be realized that acceleration schemes are not beneficial for all situations. Kouchaksaraei *et al.*[42] found that for the virtual transcoder VNF, when the input video of the transcoder has low resolution or high bitrate characteristics, using a GPU does not have advantages in performance and cost. Lopes *et al.*[43] believed that not all VNFs are suitable for FPGA acceleration.

3) *Efficient Schemes Require an Effective Combination of the Aforementioned Technologies.* We have analyzed the latest research directions of network acceleration technologies in terms of hardware offloading, kernel bypass, virtualization performance optimization, and SFC deployment. In more complex scenarios, these technologies usually only address part of the acceleration chain and need to be combined for better acceleration effects. For example, hardware acceleration technologies FPGA and sNIC leverage eBPF for task offloading[14, 44]. In the virtualized environment, using the SRIOV + DPDK method to bypass the virtualization layer and the kernel layer is a very effective scheme for deploying VNFs[29].

3 Network Cloudification

There is an escalating demand among enterprises for the hosting and dynamic management of middle-box services within public cloud infrastructures. This

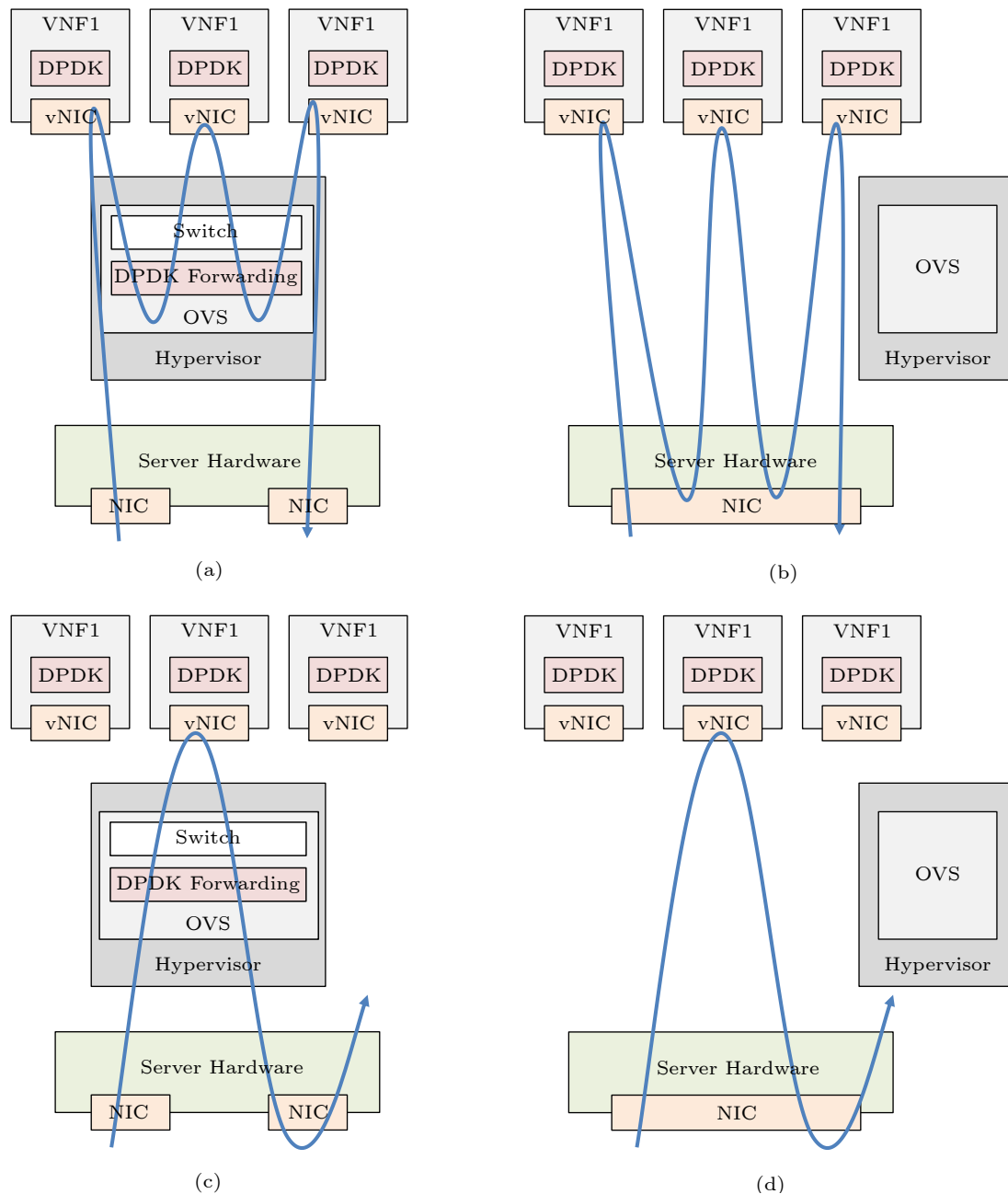


Fig.6. East-west traffic and north-south traffic comparison. (a) East-west traffic: OVS + DPDK. (b) East-west traffic: SRIOV. (c) North-south traffic: OVS + DPDK. (d) North-south traffic: SRIOV.

trend is driven by the objective to capitalize on the benefits traditionally associated with NFs in on-premises deployments^[45]. The main goal of network cloudification is to effectively deploy NFs to cloud while empowering these functions with cloud services.

3.1 Main Challenges

Network cloudification is designed to facilitate a deeper convergence between cloud and network technologies. The architecture is specifically tailored to

augment the utilization of cloud capabilities by network services and to enable cloud infrastructures to accommodate the varied requirements of distinct network services.

- The telecom network and cloud system “cannot” be integrated directly. The UNIFY project, an initiative funded under the EU’s FP7 framework, aims to merge cloud systems and carrier networks within a single architectural framework^[46]. The challenge to UNIFY involves the project’s failure to effectively address and integrate the unique characteris-

tics and operational differences between cloud technologies and network technologies. Additionally, there is a lack of clarity regarding the specific roles and responsibilities that should be allocated to cloud service providers and network operators in the network cloudification process. This oversight complicates the implementation and can hinder the seamless integration of these two critical infrastructures, potentially leading to inefficiencies, mismanagement, or underutilization of the integrated system.

- Cooperation between network operators and cloud service providers is a promising direction, but is not converged yet. Drawing on the latest practices from leading cloud providers, including Azure and AWS, we have developed a novel architecture for network cloudification, as depicted in Fig.7. While Radio Access Network (RAN), data plane VNFs, operations support systems/business support systems (OSS/BSS), and other telecommunications services employ telco cloud infrastructures for integrated hosting, this setup presents significant challenges. The infrastructures primarily relies on the complex integration of essential technologies from network virtualization, which can be technically challenging and prone to compatibility issues. Additionally, while communication between telecommunications operators and cloud service providers is expedited through direct connect, ensuring seamless and secure data exchanges between data and control planes remains a substantial operational challenge. This complexity is compounded as we aim to enhance telecommunications network services through the extensive services available on public cloud platforms, which can introduce

scalability and reliability concerns. These challenges highlight the intricate balance required to effectively implement network cloudification while maintaining service quality and operational efficiency.

Essential technologies pivotal to the implementation of network cloudification include four aspects as follows.

- *Resiliency and Scalability*: guaranteeing network services to sustain high performance and continuous availability, even amid fluctuating demands;
- *Telco Cloud Platform*: integrating diverse network services and providing enhanced flexibility in data deployment and service management;
- *Cross-Layer Operation*: facilitating seamless interactions between different layers of the network architecture, such as the physical layer, virtualization layer, and application layer, and consequently improving the efficiency and effectiveness of network operations;
- *Reliability*: ensuring that network services remain consistently accessible and dependably embedded through redundant configurations and robust failover mechanisms, thus minimizing downtime and reducing service disruptions.

3.2 Resiliency and Scalability

Resiliency in network cloudification pertains to the system's capacity to sustain and deliver an acceptable level of services in the face of faults and operational challenges. Han *et al.*[47] and Tola *et al.*[48] explored various models and strategies to bolster the resilience of network cloud environments. These

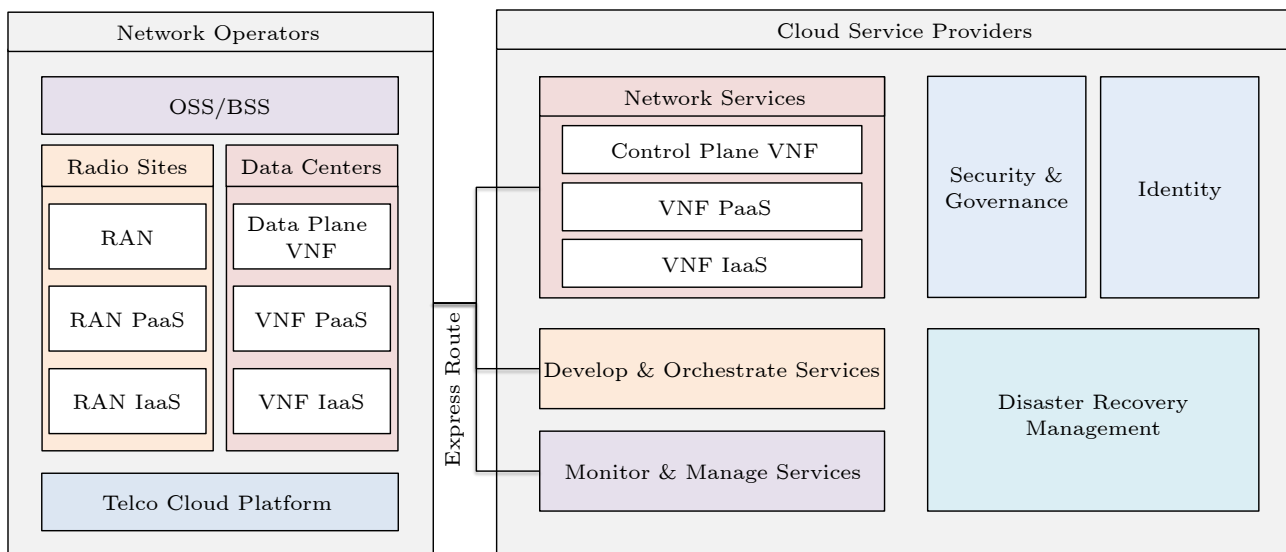


Fig.7. Network cloudification deployment architecture diagram.

strategies include the adoption of cloud-native principles that inherently enhance resilience through a microservices architecture, facilitating the independent scaling and robustness of individual NFs. Furthermore, Duan *et al.*[49] addressed another facet of resiliency by utilizing the actor model to dynamically manage the lifecycle and state of VNFs. This approach augments fault tolerance by localizing faults to individual actors, thereby averting system-wide failures and enabling swift recovery and redeployment of VNFs.

Scalability, a vital attribute of network cloudification, determines its capability to dynamically adjust fluctuating loads and to modulate resources as necessary without impairing service quality. Liu *et al.*[50] introduced a mechanism called ScaleFlux that enhances the scalability of stateful VNFs. By effectively managing state sharing and synchronization, ScaleFlux enables the NFV infrastructure to dynamically modulate its resources, adeptly addressing load variations with minimal impacts on performance. Wu *et al.*[51] further detailed intricacies of scaling stateful NFs and introduced innovative abstractions such as NFLOW and a match-value table, which facilitate the management of stateful interactions in a scalable manner. These abstractions aid in decoupling data flow management from state management, which has traditionally posed significant challenges in the scalability of NFV systems.

Resiliency and scalability in network cloudification are intricately linked with the state management. Stateful functions present unique challenges in both domains, necessitating robust strategies for state consistency and recovery to maintain resilience, along with efficient state distribution and synchronization mechanisms to ensure scalability. Additionally, the integration of cloud-native architectures and technologies such as containers and microservices, as discussed in several papers, further bolster the scalability and resilience of NFV. These technologies provide mechanisms for rapid scaling and enhance fault isolation, which are indispensable for constructing resilient network services.

Moreover, the incorporation of autonomous networking technologies enhances these attributes by enabling self-management, automated recovery, and dynamic resource allocation. These capabilities allow networks to adapt in real time to changing conditions, further improving resilience and scalability. Techniques such as machine learning algorithms can opti-

mize network performance and automate decision-making processes, while AI-driven analytics facilitate proactive fault detection and resolution.

In summary, the resilience and scalability of network cloudification are pivotal to its success and widespread use. By embracing principles from distributed systems, cloud-native technologies, and novel data management abstractions, network cloudification is well-positioned to meet the demands of contemporary network environments that require both high availability and capacity to scale seamlessly in response to variable network conditions.

3.3 Telco Cloud Platform

The telco cloud platform represents a transformative approach in the telecommunications industry, leveraging the principles of network virtualization to enhance the flexibility and efficiency of network services. At its core, the telco cloud is built upon the concept of abstracting NFs from the underlying hardware, allowing for the dynamic provisioning and scaling of network resources. This shift is largely driven by the adoption of technologies such as SDN and NFV[52]. SDN decouples the network control and forwarding functions, enabling centralized network management and increased programmability. NFV further complements SDN by virtualizing entire classes of network node functions into building blocks that can be connected or chained to create communication services. Together, these technologies form the backbone of the telco cloud, provide a robust framework that supports a wide array of telecommunications applications and services with reduced capital and operational expenditures.

The architectural design of telco clouds is inherently aimed at supporting the rigorous demands of telecom services, which include high reliability, low latency, and the ability to handle enormous volumes of data traffic. To meet these demands, telco clouds incorporate a distributed architecture that spans multiple geographical locations to bring computational resources closer to the edge of network[53]. This proximity enhances the performance of applications by reducing latency and bandwidth usage, which is crucial for emerging technologies such as 5G, IoT, and edge computing. The use of microservices in the telco cloud further enables modular and flexible service deployment, allowing telecom operators to swiftly adapt to changing market needs or technological advances.

ments. Each microservice operates independently, which simplifies updating and scaling specific parts of a system without affecting the whole, thereby enhancing the overall agility and resilience of the network.

Moreover, the transition towards a telco cloud environment is pivotal for the integration of third-party applications and services, breaking traditional silos between connectivity providers and application developers^[54]. This integration fosters an ecosystem where new services can be rapidly developed, tested, and rolled out, significantly accelerating time-to-market and innovation. For instance, in a telco cloud, VNFs can be seamlessly combined with services from startups or open-source communities, creating versatile and powerful solutions that drive forward the digital transformation. As telco clouds evolve, they increasingly adopt cloud-native principles, including containers and serverless computing frameworks, to maximize operational efficiencies and service performance. This evolution highlights the telco cloud's critical role in not only meeting the current demands of network management and service delivery but also setting a stage for future advancements in telecommunication infrastructures.

3.4 Cross Layer Operation

Firstly, in the domain of network cloudification, the cross layer operation commences with the continuous monitoring of both physical and virtual network components^[55]. Tools such as OpenStack's Ceilometer and Monasca^⑦ have become essential in this domain, providing detailed metrics for resource utilization that assist administrators in assessing the health and efficiency of their networks. These tools are capable of monitoring a variety of metrics, ranging from bandwidth usage and packet delays to more sophisticated indicators like VNF performance. Such comprehensive monitoring is critical not only for maintaining service quality but also for ensuring the operational efficiency of the underlying infrastructures^[56]. Moreover, the integration of SDN facilitates more granular network control and monitoring, enabling the retrieval of real-time data from both physical and virtual switches and routers, essential for immediate performance adjustments and long-term strategic planning^[57].

Secondly, the cross layer operation within a cloud-

ified network environment entails the analysis of collected data to detect and rectify potential issues before they affect users^[58]. This proactive approach is supported by advanced analytics and machine learning algorithms capable of predicting potential failures or bottlenecks based on both historical data and real-time inputs. For example, dynamic methods for fault detection in NFV environments are under development to supersede traditional static threshold-based systems. These methods utilize statistical models and machine learning algorithms to identify anomalies and unusual patterns that may suggest underlying issues. Such capabilities enable network operators to transition from reactive management to proactive management, thereby enhancing service reliability and user satisfaction.

Lastly, the orchestration and automation of NFs are crucial in optimizing the cross layer operation^[59]. Orchestration platforms, integral to the MANO framework, can automate the deployment and scaling of NFs based on current network conditions and anticipated demands. This automation ensures that resources are allocated efficiently, minimizing waste and enhancing service responsiveness. For instance, if VNF requests become larger, the orchestration platform can autonomously initiate scaling procedures to deploy additional instances of the VNF, thus maintaining service quality without manual intervention.

In conclusion, the cross layer operation in network cloudification is centered around sophisticated monitoring and analytics, proactive problem resolution, and highly automated orchestration. These elements amalgamate to forge a resilient, efficient, and self-optimizing network capable of meeting the demands of modern digital applications and services. As network technologies continue to evolve, the tools and strategies for cross layer operation must also adapt to ensure that networks remain robust in the face of changing technologies and market demands.

3.5 Reliability

In the domain of network cloudification, ensuring reliability is a critical concern as networks evolve toward cloud-based infrastructures. The utilization of shared physical resources in such cloudified networks introduces vulnerabilities, wherein a single point of failure could potentially impact multiple NFs and services simultaneously. Furthermore, the dynamic in-

^⑦<https://www.openstack.org/>, May 2025.

stantiation and termination of virtual functions, a hallmark of NFV environments, complicate the application of traditional reliability strategies. These challenges underscore the need for a comprehensive reevaluation of reliability assurance methods within these increasingly complex and dynamic network environments.

To mitigate these reliability concerns, a variety of strategies have been developed and deployed. Redundancy, as a fundamental approach, employs several models including 1:1, N+1, and geographical redundancy to protect network operations^[60]. These models ensure the availability of backup resources that can seamlessly assume control in the event of a failure, thereby minimizing service disruptions. Additionally, the deployment of enhanced protection mechanisms, which optimizes resource allocation while providing necessary backups, effectively balances the objectives of maintaining high reliability and managing resource costs.

Advanced fault management and recovery strategies play a pivotal role as well. Employing sophisticated monitoring tools and predictive analytics allows for the early detection and mitigation of potential issues before they result in service degradation. Automated recovery protocols facilitate rapid restoration of services, equipped with mechanisms to reroute traffic or dynamically instantiate new function instances. Such proactive measures are imperative in maintaining service continuity and adhering to the stringent reliability requirements demanded by contemporary network services.

Moreover, the design and implementation of SFCs with built-in resilience are essential^[61]. The integration of redundancy directly within SFCs and the strategic diversification of the physical resources over which these functions are distributed significantly bolster the overall robustness of the network. Policy-based management further augments reliability by enforcing standards and automating decision-making processes by concerning resource allocation and backup activation. As network cloudification continues to advance, ongoing enhancements in technologies and management strategies are crucial in ensuring that these cloud-based networks are not only efficient and scalable but also robust and reliable.

3.6 Several Observations

The synthesis of the subsections discussed reveals a coherent vision for network cloudification, focusing

on enhancing the resilience, scalability, and efficiency of network services through advanced technological integration and strategic architectural design. Observations from the discussion highlight several critical aspects.

- *Convergence of Technologies.* Network cloudification leverages the synergy between cloud computing and network technologies to create a robust framework capable of supporting dynamic NFs and services. The integration of SDN, NFV, and cloud-native principles significantly enhances the flexibility, scalability, and efficiency of network operations, allowing for more agile responses to varying demands and operational conditions.

- *Enhanced Resilience and Scalability.* The focus on resilience in network cloudification is well justified, given the critical nature of network services. Techniques such as the deployment of microservices architectures and the adoption of cloud-native principles were proved effective in isolating faults and enhancing fault tolerance. Similarly, scalability is addressed through innovative mechanisms like ScaleFlux and the use of stateful abstractions, which ensure that NFs can dynamically adjust to load variations without sacrificing performance.

- *Strategic Implementation of Telco Cloud Platforms.* The transformation within the telecommunications industry through telco cloud platforms underscores a major shift towards virtualization and service flexibility. This approach not only reduces capital and operational expenditures but also enhances the capability to manage and scale network resources efficiently. The distributed architecture of telco clouds, essential for handling high data volumes and reducing latency, is particularly advantageous for implementing advanced technologies like 5G and IoT.

- *Proactive Cross Layer Operation.* The proactive management strategies in cross layer operation, facilitated by comprehensive monitoring and advanced analytics, enable early detection and resolution of potential issues. This shift from reactive network management to proactive network management enhances overall service reliability and user satisfaction, highlighting the importance of intelligent data analysis and automated orchestration in maintaining high service quality.

- *Reliability in Dynamic Environments.* Addressing reliability in cloudified network environments involves rethinking traditional strategies to accommodate the dynamic nature of virtual functions. The im-

plementation of redundancy models and advanced fault management protocols ensures high reliability and service continuity, even in the face of network changes and potential failures.

In conclusion, network cloudification represents a significant technological evolution aimed at optimizing network service delivery and management. As this field continues to develop, it will likely introduce more sophisticated strategies and technologies that further enhance the resilience, scalability, and efficiency of network services. The ongoing research and development in this field are crucial for ensuring that network infrastructures can meet the increasingly complex demands of modern digital applications and services, thereby setting a new standard for telecommunications and network service providers.

4 Network Service Orientation

The Everything-as-a-Service paradigm has achieved great success in the field of cloud computing. Service models such as IaaS, SaaS, and PaaS offer flexible and scalable resource usage patterns, helping users reduce initial investment, operational costs, and the requirement on research and development ability, thus enhancing business agility and innovation capabilities^[62]. The decoupling of NFs from dedicated hardware has also introduced the concept of service orientation approaches. Initially, it was mainly proposed in large-scale cloud data centers, providing better support for service-based cloud delivery through Network-as-a-Service (NaaS), meeting the scalability, performance, security, and reliability of networks^[63]. NFV was also designed with service orientation principles, proposing NFV Infrastructure-as-a-Service (NFVIaaS), Virtual Network Function-as-a-Service (VNFAaaS), and Virtual Network-as-a-Service (VNaaS)^[62].

Taking the mobile network as an example, compared with 4G networks, one of the most important features of 5G networks is the introduction of SBA. Combined with the characteristics of mobile core networks and the development trend of the technology, NFs are divided into several reusable services. The goal is to achieve efficiency, softwarization, and openness of 5G systems. At the same time, standardized programming interfaces (APIs) expose network capabilities to external programs, giving rise to network applications (NetApps)^[64], and providing more room

for value-added network services. Based on the advantages brought by the use of an SBA architecture in 5G core networks, next generation of networks, such as 6G networks, will further promote the evolution of network architectures towards a fully service-oriented approach, achieving a more flexible holistic service orientation architecture with full coverage and full applications^[65].

4.1 Main Challenges

Network virtualization enables the deployment of NFs on COTS servers, and network cloudification further leverages the cloud's elasticity, high reliability, operation and maintenance, and security features to support NFs. These provide firm ground for better realization of network service orientation. However, the complexity of network services, the evolution of network architectures, and the requirements for compatibility, network stability, and reliability still present us with a series of challenges. The goal of network service orientation is to provide more elastic, scalable, flexible, customizable, and rapidly deployable network services. How to decompose and combine NFs based on a service-oriented architecture, and how to better implement network services (such as configuration management) under a new architecture are key issues.

- *Service Models are not Well Defined yet.* Drawing on a value demonstrated by cloud-native and microservices technologies in IT applications, networks have also adopted related technologies when introducing service-oriented architectures. However, in the process of providing virtual NFs, equipment vendors often bundle microservices-based NFs with cloud-native infrastructures in a closed "black box" delivery model. A lack of unified and effective service models, and the need for customized cloud-native infrastructures, are the main reasons and challenges.

- *Supporting Real-Time Configurations in Large Scale Network is Non-Trivial.* Network configurations are an important way to implement network business functions. In a service-oriented architecture, ensuring the timeliness, consistency, and effectiveness of configurations among different replicas of the same service and different instances of various services is crucial. This challenge is particularly significant in large-scale carrier networks because it requires handling of a vast amount of data, with extremely high demands for

configuration accuracy, response speed, and concurrency.

In response to the above challenges, as shown in Fig.8, we systematically explain the related work in two aspects.

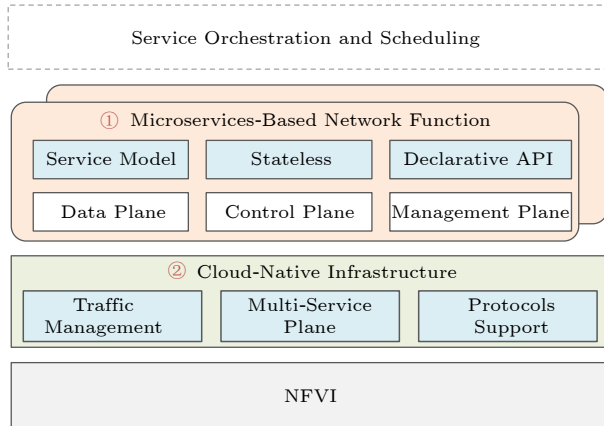


Fig.8. Network service orientation architecture diagram.

- *Application Layer.* It mainly optimizes NFs, systematically decomposes and combines NFs through service models, decouples state information with stateless applications for better scaling, and achieves real-time configuration of large-scale networks through declarative APIs.

- *Infrastructure Layer.* The evolution of NFs towards a service-oriented architecture requires enhancements to general cloud infrastructures, as its unique characteristics, such as multi-network screens, dense traffic, low-latency transmission, and support for specific protocols, often cannot be met by common cloud infrastructures, necessitating enhancement for better support.

4.2 Microservices-Based Network Function

The IT application architecture has evolved from the monolithic application architecture, Service-Oriented Architecture (SOA), microservice architecture, to serverless architecture. The granularity of services has become finer, and the degree of decoupling has increased, providing more flexible development, iteration, deployment, and scaling capabilities. Viewing NFs as a complex IT application, they are also developing in this direction, with NFs based on microservice architecture becoming a trend^[66]. The complexity of NFs transforms microservices challenging, being

reflected in the service model, statelessness, and declarative configuration.

Service Model. The service model describes the boundaries of microservices and how to combine microservices into a network service, a process that is illustrated in the diagram. Determining the optimal microservice boundary is one of the key factors affecting service quality^[67]. Excessive service granularity affects the efficiency of module reuse, while too fine granularity introduces additional complexity and overhead. There are some mature methods for microservice design, such as Domain-Driven Design (DDD)^[68], which solves complex business problems by building a rich domain model, ensuring the consistency of the business model and the code model, and achieving high cohesion within the service and low coupling externally. Rademacher *et al.*^[69], in response to the limitations of DDD as an informal modeling language, proposed a UML profile oriented method to achieve model verification and transformation. Unlike DDD, which is based on expert experience, Roman *et al.*^[70] designed a data-centric microservice splitting method, which automatically constructs business context splitting points through pattern analysis, clustering, and other data mining methods, identifies themes and divides boundaries. Telecommunication operators use domain models to describe complex architectural design^[71]. Currently, there is no systematic proposal for the design method of microservices, and atomic functional partitioning is usually based on historical modules. However, there are relevant achievements on how to combine these atomic capabilities (microservices) into network services, especially in the 5G slicing scenario, where different users require different services.

The Communication Service Management Function (CSMF) defined by 3GPP[®] is a key component in the 5G network slicing architecture. The main task of CSMF is to act as a converter between communication service requirements and network slicing requirements, while also providing service design and cataloging, as well as lifecycle management of service instances. The Product Service Resource model (PSR), defined by TeleManagement Forum (TMF), is a service framework used in the telecommunications industry. It divides business processes into three layers: Product, Service, and Resource. Resource provides network infrastructures, Service provides assem-

[®]3GPP. Telecommunication management; Study on management aspects of communication services. 2018. <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3555>, Mar. 2025.

bleable network capabilities, and Product provides the final solution to customers. Both CSMF and RSP essentially define the mapping from requirements/products to services. Services can be resource services or abstracted general microservices. Through mapping, standardization, and open sharing, they achieve flexible assembly, rapid loading, and activation of NFs. These service models help to achieve flexibility and scalability in telecommunications services, enabling operators to manage and provide services more efficiently.

Network Functions Stateless. NFs are categorized into two types: stateful and stateless. Stateful NFs persist the state information required for NF operation locally, tightly coupled with the processing component; stateless NFs, on the other hand, decouple the processing component from the state information, with the application itself not storing the state information. It is clear that stateless NFs possess better scalability, flexibility, and fault tolerance, which can better leverage the advantages of a service orientation architecture^[72], and are naturally compatible with cloud-native platforms and serverless platforms. NFs design is typically closely related to specific states and has characteristics such as high traffic intensity and low latency requirements. Frequent synchronization and data consistency make the stateless transformation face many challenges. Kulkarni *et al.*^[73] analyzed the comparison between stateful NFs and stateless NFs in terms of elastic scaling and fault recovery, and believed that although stateless NFs are promising, they face issues such as execution orders and consistent state updates, and cannot achieve line-rate packet processing, necessitating further optimization of a persistent storage layer. In response to this issue, Szalay *et al.*^[74, 75] proposed a complete state layer to minimize the inter-host communication caused by state synchronization (including state access time and network traffic), and verified it in an IP multimedia subsystem, capable of recovering the real-time call records of thousands of mobile users within a few seconds, and only requiring half the resources needed for stateful design. In addition, Kulkarni^[76] proposed a stateless-based 5G systems to reduce the latency added to a transmission process of stateless information in NFs due to data serialization and deserialization. Goshi *et al.*^[77] proposed a Procedure-Aware Stateless System for 5G to address the perfor-

mance loss caused by frequent state transactions.

Declarative Configuration. Declarative approaches describe a desired end state, allowing the system to automatically determine how to achieve it, while imperative approaches require specifying the concrete steps to reach these states. Leveraging the successful experience of Kubernetes' declarative APIs, the declarative philosophy has been widely applied and holds greater application value in the NFV domain (in terms of the complexity and difficulty of configuration management), representing an important trend towards network automation and simplified management. Through declarative configuration, complex network management tasks are simplified, improving the readability and maintainability of configurations and reducing the risks associated with manual configuration errors (which are considered one of the significant causes of network failures). Wu and Wang^[78] and Stolyarov^[79] both proposed managing NFs using declarative methods, including TOSCA template-based and Kubernetes integration approaches. The open-source project Ligato (which can be understood as a proxy for VPP) also utilizes Kubernetes to implement declarative configuration and statelessness^⑨. Additionally, declarative configuration is a key method for implementing intent-based networking, ensuring that the user's network intentions are met by automatically achieving the end state. This is reflected in work done in [80, 81]. Of course, declarative configuration requires modifications to existing NFs, which brings additional workload, and although it can be accomplished with mature declarative frameworks (such as Kubernetes), we believe it is worthwhile, enabling better management of network services.

4.3 Cloud-Native Platform Support

Compared with monolithic applications, service orientation architectures offer greater flexibility, but they also incur additional communication overhead, especially in terms of performance. This overhead grows rapidly along with the number of services^[82]. For traffic-intensive NFs, this overhead is even more significant and cannot be ignored. Efficiently directing and managing traffic between services presents a new challenge. At the same time, we observe that due to the special requirements of NFs, such as support

^⑨Ligato: An open source NFV platform. <https://github.com/ligato>, 2019. Mar. 2025.

for multiple network planes and telecommunications protocols, new demands are being placed on cloud-native infrastructures.

Network Service Mesh (NSM). In a microservices architecture, applications are broken down into multiple microservices. Inter-service communication, management (service discovery and registration), observability, reliability, and other aspects need to be addressed. The service mesh serves as an infrastructure layer for communication between microservices, shielding services from this complexity. Compared with the SFF module in the SFC architecture, service mesh proxies offer more functionality but also introduce performance issues. There is already some work studying how to use service mesh to facilitate communication between network services in a cloud-native environment. Dab *et al.*^[83] proposed an optimized network-aware complex balancing strategy that achieves efficient traffic steering by providing the best-performing VNF replicas for network services. Sedghpour *et al.*^[84] and Yang *et al.*^[85] found that accelerating the service mesh with eBPF is an effective method, which is consistent with the approach we discussed in Subsection 2.3.

Bittar *et al.*^[86] and others have verified that deploying network services with the NSM framework is limited and greatly related to the functionality and features of NSM. On this basis, Duong *et al.*^[87] designed a Cilium-based service mesh to enhance the security and observability of 5GC. Compared with IT applications, NFs require specialized communication protocols to ensure efficient and reliable information exchange. Taking the 5GC network as an example, protocols such as GTP and PFCP are needed for communication between network elements, which are not supported by the service mesh. Aldas *et al.*^[88] leveraged the programmability of the service mesh to extend the Envoy proxy to meet the communication protocol requirements of 5GC, while minimizing the performance impact of this extension. In addition, addressing multi-cluster, cross-domain service communication^[89], and providing sufficient network isolation mechanisms to meet security requirements are also issues that need to be resolved.

There is no clear conclusion on whether to use service mesh for service governance after the service orientation NFs. Essentially, it is a trade-off between performance and functionality (load balancing, observability, security, and traffic control). However, looking at the trend of technological evolution, the

service mesh is necessary. The future technological direction is whether to enhance the service mesh oriented to IT scenarios or to customize service mesh for VNFs, which is also a direction that needs to be studied.

Multi-Service Plane. Communication between NFs relies on specific interfaces, which require static IP addresses to ensure stable network connections between different functions. In addition, some NFs need to be installed on multiple interfaces. Therefore, cloud-native infrastructures must provide static IP addresses for communication interfaces, as well as support for multiple network planes. It is clear that an access method based on static IPs will limit the scaling and migration of service instances, making traffic management more difficult. Introducing multiple service planes can better utilize the advantages brought by service orientation, such as elasticity, load balancing, and high availability. The multi-service plane model is shown in the Fig.9.

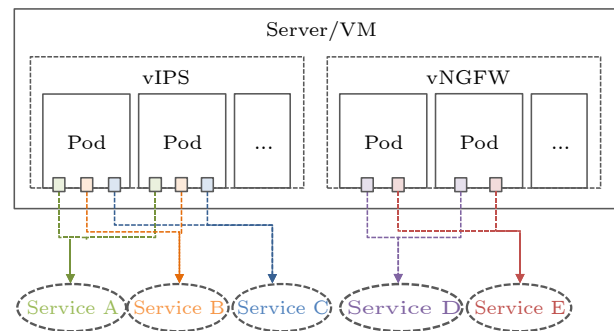


Fig.9. Multi-service plane model.

4.4 Observations on Application of Serverless Computing in Service Orientation

With the rise of serverless architecture, its characteristic of allocating computing resources on demand has brought a new perspective to the design and implementation of NFs. The serverless architecture allows developers to build and run applications and services without managing the complexity of servers, offering potential advantages for the dynamic scaling and cost-effectiveness of NFs. Whether NFs should further adopt a serverless computing paradigm to achieve more economical and efficient benefits has also sparked discussions in academia. Aditya *et al.*^[90] believed that compared with traditional virtual machines and containers, serverless systems have capabilities such as low latency, scalability, low cost, and

rapid deployment, without the need for developers to perform additional configurations, and can be applied to the NFV domain, exploring methods for building adaptive and scalable networks using serverless technology. Savi *et al.*^[91] held a more conservative view, and they tested the performance of serverless platforms carrying NFs, affirming that serverless computing can bring some benefits, but compared with the system overhead, the overall effect may not necessarily be better. It is suggested that short control plane VNF requests can use serverless computing, while traffic-intensive data plane VNFs should avoid it, which also matches the scenarios suitable for serverless computing. In addition, the work of using serverless computing to carry NFs in scenarios with relatively limited resources, such as edge computing and the IoT, is also advancing^[92, 93].

5 Future Directions

As infrastructures continue to be upgraded, the application scenarios of NFV are also expanding. Application scenarios of NFV have already moved from cloud services such as virtual private networks and server load balancers, to security fields like intrusion detection systems and firewalls, and to core networks of operators such as vBRAS, vIMS, and 5GC. Recently, the application of NFV has also been extended to scenarios like fiber access networks^[94] and wireless access networks^[95]. Looking to the future, with the introduction and breakthroughs of cutting-edge technologies such as new types of acceleration hardware, artificial intelligence, and cloud network convergence, we have reason to believe that NFV will unleash greater potential and promote the continuous improvement of new-generation network capabilities. We will next explore some promising directions.

5.1 Standardization

Traditional NFs are relatively closed systems, and achieving decoupling from hardware resources, joint deployment of virtual network elements, and management and orchestration requires multi-vendor collaboration. Standards are the primary and important force in driving this process.

At the end of 2012, the establishment of the first NFV standard organization, ETSI NFV Industry Specification Group (ISG), marked the starting point of the network's evolution and transformation to-

wards virtualization. Telecommunication operators and equipment manufacturers are accustomed to developing products and putting them into commercial use in accordance with common standard processes, while IT manufacturers and open source communities advocate the concept of "code first". From NFV's Release 1 in 2013 to Release 5 in 2021, a total of five major versions have been released, going through the processes of reference architecture design, NFV interoperability interfaces, operability enhancement, cloud-based deployment and automation, automated operation and maintenance, and strengthened support for cloud-based deployment. The efforts are dedicated to introducing IT concepts into the CT field, gradually achieving virtualization, cloudification, and service orientation.

Other standard organizations are also conducting research in conjunction with network virtualization. In the latest work of ETSI Multi-Access Edge Computing (MEC) ISG, it has begun to support the integration of MEC and NFV, enabling MEC applications and VNFs to share the same virtual infrastructure, and NFV MANO can be used for MEC management and orchestration. This integration achieves resource sharing and optimization, improves network performance and flexibility, and expands the service scope of MEC. The 5G network architecture proposed by the 3GPP/5G-PPP community regards network slicing as the key to implement multi-tenant virtual networks. NFV has enhanced support for 5GC in its releases, enabling the rapid creation and adjustment of network slices according to specific service requirements. The consensus that the Internet Engineering Task Force (IETF) Service Function Chaining (SFC) complements NFV has been established, with network virtualization providing more possibilities for network services. The Service Function and Service Function Forwarder of SFC are typically deployed in virtualized forms in practice, such as virtual intrusion prevention systems, virtual firewalls, and virtual routers. In the recently proposed hierarchical architecture of SFC, integration with NFV can provide more flexible cross-domain services. In addition, Metro Ethernet Forum Lifecycle Service Orchestration attempts to achieve flexible service orchestration across multiple provider networks and different technical domains.

In the field of networking, standardization serves as a crucial bridge for technologies to transition from research to application. The virtualization, cloudifica-

tion, and service orientation of networks require the concerted efforts of operators, server and equipment manufacturers, and cloud service providers to ensure compatibility and interoperability among different systems and components. At the same time, the introduction of emerging technologies poses higher demands on standardization work, such as how to effectively integrate cutting-edge technologies like AI and cloud-native into existing architectures, and how to achieve the organic combination of open-source projects with industry standards. An increasing number of standardization organizations are actively responding and are committed to advancing the standardization process in these key areas, which will be an important direction.

5.2 Customizable Computing

The ideal state of network virtualization is to have the performance of dedicated hardware while maintaining the flexibility and cost-effectiveness of general-purpose hardware. This is an extremely challenging goal. We believe that customizable computing is one of the most promising technologies to achieve this vision. Customizable computing achieves this by specializing the hardware for specific application areas, enabling performance and energy efficiency improvements 10–100 times higher than general-purpose processors^[96], while also giving hardware greater flexibility to dynamically change its structure and functions according to different computing tasks.

Customizable computing is mainly divided into loosely coupled accelerators (LCAs), FPGAs, and coarse-grained reconfigurable arrays (CGRAs). Currently, accelerated hardware such as smart network cards and DPUs are primarily built on technologies like application-specific integrated circuits (ASICs) and FPGAs, but they still struggle to balance performance and programmability. CGRA, also known as a composite accelerator, combines the high performance of LCAs with the high reconfigurability of FPGAs, providing performance close to dedicated hardware in network virtualization environments while maintaining a certain level of flexibility. Despite its obvious advantages, CGRA also faces some challenges, including compiler design, resource management, control flow processing, hardware design complexity, the establishment of a software ecosystem, and the complexity of application development.

5.3 Heterogeneous Infrastructures

The application of customizable computing leads to the emergence of a large amount of heterogeneous hardware. One of the biggest transformations in data centers over the past decade has been the rapid growth of heterogeneous computing^[97]. In the evolving landscape of network virtualization, the trend towards employing heterogeneous infrastructures is gaining significant traction. This shift is primarily driven by the diverse requirements of new application scenarios including video processing and cloud radio access networks (CloudRANs). As these applications demand increasingly specialized computational capabilities, the deployment of a variety of heterogeneous hardware resources within NFV systems is becoming inevitable. For instance, high-performance VNFs are now requiring advanced acceleration hardware such as DPUs, GPUs, network processing units (NPUs), FPGAs, and AI-specific ASICs. These resources are essential for meeting the intensive computational demands posed by modern network services and applications.

Based on heterogeneous infrastructures, we believe that the design concept of soft-hardware integration (achieving the flexibility of software and the high performance of hardware through deep collaboration between software and hardware) is an important direction for the future. The core lies in the programmability of hardware and the hardware awareness of software. Through the programmability of hardware, network devices can dynamically adjust their functions according to the needs of software, thereby improving resource utilization and performance. At the same time, software needs to be able to perceive the status and capabilities of hardware in order to use hardware resources more effectively and achieve performance optimization. This requires an in-depth understanding of the interaction between software and hardware, and exploring how to effectively combine them to achieve optimal performance. For example, how to intelligently determine which parts should be offloaded to hardware, how to effectively schedule hardware and CPU resources to achieve load balancing, and how to quickly move data between the CPU and hardware are issues that still need to be resolved.

Different heterogeneous hardware has varying processing speeds and performance advantages for NFs, and the computing frameworks and programming languages they each support also differ, which poses sig-

nificant challenges for upper-layer applications. On the one hand, there is a need for a heterogeneous resource management platform that provides a unified way to manage and coordinate these different types of resources, thereby simplifying the management process and improving resource utilization. On the other hand, a unified programming framework is needed to abstract the heterogeneity of the hardware, which can intelligently allocate the most suitable resources based on the characteristics of workloads.

5.4 Multi-Cloud and Distributed Cloud

Multi-cloud and distributed cloud architectures are major trends in the evolution of network cloudification. A multi-cloud model leverages resources from multiple cloud service providers, thus enabling the telco cloud to achieve higher levels of redundancy and geographic diversity. This diversity is crucial for network services that require high availability and resilience against regional failures. Multi-domain orchestration facilitates a distributed management approach where services can be seamlessly provisioned and managed across different cloud environments, regardless of the underlying provider^[98]. This orchestration not only optimizes resource utilization but also ensures that service levels are maintained across disparate cloud domains.

Distributed cloud models extend this concept by decentralizing computational resources and services closer to end-users, which significantly reduces latency and enhances the performance of NFs. Distributed cloud architectures support a wide range of applications, from mobile edge computing to large-scale IoT deployments, by enabling dynamic resource allocation and real-time service orchestration^[99]. The integration of NFV with distributed cloud frameworks inherently supports scalability and agility, allowing network operators to adapt to changing demands quickly. Moreover, the framework proposed by Abu-Ledeh *et al.*^[100] emphasizes the importance of interoperability among different cloud providers, which is crucial for the successful deployment of NFs in a multi-cloud scenario, ensuring that services can be managed efficiently across various platforms without vendor lock-in issues.

5.5 Autonomous Network

With the continuous development of the artificial

intelligence technology, it plays an increasingly important role in networks^[101]. The concepts of cloud-native DevOps and GitOps can be applied to enhance the end-to-end automation capabilities of systems and support the automation of NFV systems from single lifecycle management to service design, testing, and integration. With the comprehensive improvement of end-to-end automation capabilities in NFV, the AI technology can be further introduced to achieve automatic decision-making and optimization through artificial intelligent operations (AIOps) and machine learning operations (MLOps) technologies.

As a declarative intent-driven approach, by introducing AI technologies, it can better understand users' needs for the network. During the network planning and deployment phase, the digital twin technology can be used to simulate and verify automated strategies. When NFs fail, fault prediction and prevention capabilities can transform a fault from a response mode to an active predictive management mode. For issues that arise during operation, automatic analysis of root causes and automatic fault location can ensure the robust operation of the network. At the same time, AI can be used for capacity planning and optimization, optimizing resource allocation, especially in edge nodes, by predicting network load changes and scheduling network resources in a timely manner, reducing operational costs.

5.6 Declarative Intent-Driven

It is evident that the complexity of virtualized network deployment has gradually increased due to the introduction of virtualization and cloud technologies (from the perspective of vertical network decomposition), as well as the overly segmented customization in actual network deployments. The declarative intent-driven approach, based on the declarative configuration mentioned in [Subsection 4.2](#), introduces the concept of intent. It provides an abstract method to convert the understanding of user intent into corresponding network behaviors, influencing the design and deployment of network infrastructures and services. Networks based on the declarative intent-driven approach are an important cornerstone for the evolution of network and service management towards a truly autonomous paradigm^[102].

The main challenges include the accurate identification of intent and the automated generation and execution of policies. The identification of intent requires advanced natural language processing tech-

niques to parse user intent and transform it into specific network operations, which necessitates the abstraction of intent into an appropriate representation model based on the knowledge of domain experts. The automated generation and execution of policies require a complex logic and rule engine to construct corresponding network policies for the parsed intent representation model, verify the effectiveness of the policies, and continuously optimize them during the execution process.

5.7 Cloud Network Convergence

Networks and computing are evolving from being independent to being integrated, with the emergence of new technologies represented by cloud network convergence. This convergence is not simply about piecing together computing resources represented by cloud computing, edge computing, etc., with network resources represented by 4G/5G, optical networks, the Internet, etc. Instead, it is about leveraging technological innovations to enable two types of information technologies, which have different characteristics and paths, to penetrate and influence each other. This ultimately leads to deep integration and innovation, merging various resources into a system that provides integrated supply, operation, and services. By constructing a new type of information infrastructures, cloud network convergence aims to empower the digital economy.

This interdisciplinary integration has stimulated innovation in the cloud network technology, with network virtualization, cloudification, and service orientation as core technologies receiving widespread attention. They are not only the core technologies of cloud network integration, but also its typical application. On the one hand, the cloud's flexible customization and rapid delivery have put forward higher demands on the network. Technologies such as elastic scaling, network programmability, quick service activation, and atomic capability service orientation are used to provide a flexible and efficient network for cloud access, within a cloud, and between clouds. On the other hand, by building a unified carrying of multi-professional virtual network elements in clouds, along with capability opening and service enhancement, the evolution of the network is accelerated.

6 Related Work

The academic community has also conducted ex-

tensive research in these areas, and a number of focused survey results have been published. This paper reviews the relevant summary literature from 2019, as can be seen in [Table 2](#).

In the field of network virtualization^[19, 103, 104, 107], research primarily focuses on two directions: virtualization acceleration technologies and virtualization management platforms. The study of virtualization acceleration technologies aims to address a significant performance gap between COTS servers and dedicated equipment, which is crucial in dealing with the rapid growth of network traffic. Meanwhile, the research on virtualization management platforms addresses an issue of lacking a unified platform to support diverse network functions. In practical applications, a management layer of NFs is often customized, which can pose obstacles in large-scale commercial deployments.

In the field of network cloudification^[62, 105, 106], there is a need to resolve issues such as security trust, operations and maintenance, and high-performance demands faced when migrating network functions to the cloud. Current practical experience indicates that directly deploying VNFs to public clouds is not feasible. Additionally, attention is given to leveraging the capabilities of the cloud to enhance network functions, such as improving the elasticity and cross-domain service capabilities of network functions through distributed cloud architectures.

In the field of network service orientation^[107–109], research concentrates on network service architectures represented by SFC, focusing on their orchestration, deployment, and reliability. By optimizing placement strategies, a goal is to mitigate traffic fluctuations and high-latency issues that may arise from network function virtualization, thereby providing more reliable, scalable, and high-performance services.

Undoubtedly, these literature pieces have conducted in-depth research on specific topics and provided unique insights. However, they still lack comprehensiveness and a macroscopic perspective, failing to fully present an overall development trend of NFV. Moreover, they lack a systematic discussion on the technical challenges and optimization strategies in accelerating large-scale applications of NFV. For instance, the virtualization acceleration literature proposes classification methods that lack practical guidance in real-world application. The literature on network cloudification lacks discussions on how to utilize cloud services and the high reliability of multi-AZ

Table 2. Comparison of Related Surveys and Their Scope

Reference	Scope (Network)			Contribution
	Virtua- lization	Cloudi- fication	Service orientation	
Fei <i>et al.</i> ^[19] (2020)	✓	✗	✗	Analyzing the performance challenges inherent in COTS servers, this paper proposes a novel method for accelerating classification, identifying the strengths and weaknesses of each category. It also discusses products and solutions that have emerged within the industry.
Huang <i>et al.</i> ^[103] (2022)	✓	✓	✗	A set of network acceleration classification methods based on the ETSI architecture has been proposed, offering a more comprehensive and intuitive view for acceleration technologies and solutions. It is believed that in the future, NFV will increasingly rely on cloud services, tending to build acceleration solutions based on cloud services.
Zhang <i>et al.</i> ^[104] (2020)	✓	✗	✗	The paper provides a comprehensive review for existing network virtualization platforms, classifying them based on functionalities they offer throughout the network function lifecycle. It elaborates on design principles of each category of platforms, offering detailed guidance for network operators and service providers in selecting a network virtualization platform.
Souza <i>et al.</i> ^[105] (2020)	✗	✓	✗	This paper primarily targets the application of network virtualization in data centers, analyzing issues related to performance, reliability, resource allocation, cost, management, and resource interoperability. It suggests that further collaboration between academia and industry is needed to achieve the practical implementation of network function virtualization.
Zoure <i>et al.</i> ^[106] (2022)	✗	✓	✓	The enterprise's migration of network functions to the cloud requires an understanding of potential network service exceptions to assess risks. This paper analyzes the impact of network service exceptions on critical service attributes, compares and categorizes existing relevant literature, and proposes future research directions.
Duan <i>et al.</i> ^[62] (2020)	✗	✓	✓	The extensive application of NFV has introduced a trend towards the cloudification of networks. This paper analyzes a current state of technology in the convergence of networks and cloud edges, as well as challenges that need to be addressed during this process, and directions for future research.
De <i>et al.</i> ^[107] (2019)	✓	✗	✓	Emerging scenarios require network services to span data centers across heterogeneous infrastructures and management domains. Research and development efforts are dedicated to enabling orchestration processes that can automate, coordinate, and manage the deployment and operation of network services.
Kaur <i>et al.</i> ^[108] (2020)	✗	✗	✓	A hierarchical classification method is proposed, providing efficient reference configurations for technologies such as VNF placement and load balancing. The paper discusses research and development challenges in deployment of SFC, assisting cloud service providers in delivering reliable, scalable, and high-performance services to their customers.
Santos <i>et al.</i> ^[109] (2022)	✗	✗	✓	In highly distributed scenarios, service chains composed of VNFs can be affected by traffic fluctuations and high latency. This paper examines the placement of SFC in distributed scenarios over the past decade, discussing placement strategies to meet service requirements of network operators.
Azadiabad <i>et al.</i> ^[110] (2024)	✗	✗	✓	VNFs introduce challenges to the reliability of network services due to resource sharing, dynamic capabilities of network services, and the elasticity of VNFs. This paper surveys existing solutions for NFV reliability, discussing gaps in current approaches and future directions for work.
Our work (2025)	✓	✓	✓	To accelerate large-scale applications of NFV, this paper systematically discusses research progresses of the past five years from three directions: network virtualization, network cloudification, and network service orientation. It analyzes current problems and challenges, and provides corresponding suggestions on how to address them. It also forecasts future directions of technological development.

to further enhance network functions. The focus of network service orientation on optimizing cost and resources through more rational orchestration and

scheduling strategies may not address current pain points, which might be more about how to transform network functions into services.

7 Conclusions

In conclusion, the systematic review presented in this paper has comprehensively explored the inter-linked domains of network virtualization, cloudification, and service orientation architectures, which collectively contribute to the innovative transformation of network infrastructures. This transformation is driven by the need for more flexible, automated, and intelligent network systems capable of supporting emerging applications and technologies such as the Internet of Things, big data, and AI.

The evolution from traditional NFs to highly flexible virtualized functions has fundamentally shifted the landscape of network design and operation. NFV has emerged as a cornerstone, enabling the decoupling of NFs from physical hardware and facilitating their deployment on cloud-based infrastructures. This shift not only enhances the agility and scalability of networks but also introduces significant challenges in terms of standardization, performance optimization, and security.

Despite the progress made in the past decade, the full potential of NFV has not yet been realized. Issues such as performance bottlenecks, security concerns, and the complexity of management and orchestration continue to hinder widespread adoption. The integration of NFV with cloud-native technologies and service orientation architectures offers promising avenues for addressing these challenges, as demonstrated by the current trends towards multi-cloud and hybrid cloud environments, and the adoption of service-based architectures.

Looking forward, the future of network architectures lies in continuous integration of these technologies to further enhance the flexibility, efficiency, and user-centricity of network services. The relationship between network virtualization, cloudification, and service orientation frameworks will become increasingly synergistic, driving the next wave of innovations in the networking domain.

Conflict of Interest Jie Wu is an editorial board member for Journal of Computer Science and Technology and was not involved in the editorial review of this article. All authors declare that there are no other competing interests.

References

- [1] Tsareva I, Doan T V, Bajpai V. A decade long view of Internet traffic composition in Japan. In *Proc. the 2023 IFIP Networking Conference (IFIP Networking)*, Jun. 2023. DOI: [10.23919/IFIPNetworking57963.2023.10186393](https://doi.org/10.23919/IFIPNetworking57963.2023.10186393).
- [2] Basta A, Kellerer W, Hoffmann M, Morper H J, Hoffmann K. Applying NFV and SDN to LTE mobile core gateways, the functions placement problem. In *Proc. the 4th Workshop on All Things Cellular: Operations, Applications, & Challenges*, Aug. 2014, pp.33-38. DOI: [10.1145/2627585.2627592](https://doi.org/10.1145/2627585.2627592).
- [3] Niemiec G S, Batista L M S, Schaeffer-Filho A E, Nazar G L. A survey on FPGA support for the feasible execution of virtualized network functions. *IEEE Communications Surveys & Tutorials*, 2020, 22(1): 504-525. DOI: [10.1109/COMST.2019.2943690](https://doi.org/10.1109/COMST.2019.2943690).
- [4] Han S J, Jang K, Park K, Moon S. PacketShader: A GPU-accelerated software router. *ACM SIGCOMM Computer Communication Review*, 2010, 40(4): 195-206. DOI: [10.1145/1851275.1851207](https://doi.org/10.1145/1851275.1851207).
- [5] Kim J, Jang K, Lee K, Ma S, Shim J, Moon S. NBA (network balancing act): A high-performance packet processing framework for heterogeneous processors. In *Proc. the 10th European Conference on Computer Systems*, Apr. 2015, Article No. 22. DOI: [10.1145/2741948.2741969](https://doi.org/10.1145/2741948.2741969).
- [6] Sun W, Ricci R. Fast and flexible: Parallel packet processing with GPUs and click. In *Proc. the 2013 Architectures for Networking and Communications Systems*, Oct. 2013, pp.25-35. DOI: [10.1109/ANCS.2013.6665173](https://doi.org/10.1109/ANCS.2013.6665173).
- [7] Jung C, Kim S, Kim Y, Yeom I. Virtualizing GPU direct packet I/O on commodity Ethernet to accelerate GPU-NFV. *Journal of Network and Computer Applications*, 2022, 206:103480. DOI: [10.1016/j.jnca.2022.103480](https://doi.org/10.1016/j.jnca.2022.103480).
- [8] Kundel R, Anderweit L, Markussen J, Griwodz C, Aboud O, Becker B, Meuser T. Host bypassing: Let your GPU speak Ethernet. In *Proc. the 8th IEEE International Conference on Network Softwarization (NetSoft)*, Jun. 27-Jul. 1, 2022, pp.85-90. DOI: [10.1109/NetSoft54395.2022.9844090](https://doi.org/10.1109/NetSoft54395.2022.9844090).
- [9] Guo L, Zhang K, Wang X S. Gaviss: Boosting the performance of GPU-accelerated NFV systems via data sharing. *IEEE Trans. Parallel and Distributed Systems*, 2022, 33(12): 4472-4483. DOI: [10.1109/TPDS.2022.3193368](https://doi.org/10.1109/TPDS.2022.3193368).
- [10] Sambasivam B, Subramanian M, Chatterjee D, Gouda M, Sethuramapandian S, Saroha Y S. Writing P4 compiler backend for packet processing engines. In *Proc. the 2022 Symposium on Architectures for Networking and Communications Systems*, Dec. 2021, pp.109-112. DOI: [10.1145/3493425.3502769](https://doi.org/10.1145/3493425.3502769).
- [11] Arikawa Y, Sakamoto T, Shigematsu S. FPGA-based radio-resource scheduler for 5G mobile in NFV environments. *IEICE Communications Express*, 2019, 8(7): 263-268. DOI: [10.1587/comex.2019XBL0048](https://doi.org/10.1587/comex.2019XBL0048).
- [12] Wang K, Chang Y, Guo Z. High performance network virtualization architecture on FPGA SmartNIC. *IEICE*

- Trans. Communications*, 2023, E106.B(6): 500–508. DOI: [10.1587/TRANSCOM.2022EBP3122](https://doi.org/10.1587/TRANSCOM.2022EBP3122).
- [13] Lettieri G, Fais A, Antichi G, Prociassi G. SmartNIC-accelerated stream processing analytics. In *Proc. the 2023 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Nov. 2023, pp.135–140. DOI: [10.1109/NFV-SDN59219.2023.10329593](https://doi.org/10.1109/NFV-SDN59219.2023.10329593).
 - [14] Abranches M, Michel O, Keller E, Schmid S. Efficient network monitoring applications in the kernel with eBPF and XDP. In *Proc. the 2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Nov. 2021, pp.28–34. DOI: [10.1109/NFV-SDN53031.2021.9665095](https://doi.org/10.1109/NFV-SDN53031.2021.9665095).
 - [15] Miano S, Doriguzzi-Corin R, Risso F, Siracusa D, Sommese R. Introducing SmartNICs in server-based data plane processing: The DDoS mitigation use case. *IEEE Access*, 2019, 7: 107161–107170. DOI: [10.1109/ACCESS.2019.2933491](https://doi.org/10.1109/ACCESS.2019.2933491).
 - [16] Michalowicz B, Suresh K K, Subramoni H, Panda D K D, Poole S. Battle of the BlueFields: An in-depth comparison of the BlueField-2 and BlueField-3 SmartNICs. In *Proc. the 2023 IEEE Symposium on High-Performance Interconnects (HOTI)*, Aug. 2023, pp.41–48. DOI: [10.1109/HOTI59126.2023.00020](https://doi.org/10.1109/HOTI59126.2023.00020).
 - [17] Barsellotti L, Alhamed F, Olmos J J V, Paolucci F, Castoldi P, Cugini F. Introducing data processing units (DPU) at the edge. In *Proc. the 2022 International Conference on Computer Communications and Networks (ICCCN)*, Jul. 2022. DOI: [10.1109/ICCCN54977.2022.9868927](https://doi.org/10.1109/ICCCN54977.2022.9868927).
 - [18] Zhang H, Zhang C, Sun Q. Considerations on cloud-network convergence. *Information and Communications Technology and Policy*, 2022, 48(11): 1–6. DOI: [10.12267/j.issn.2096-5931.2022.11.001](https://doi.org/10.12267/j.issn.2096-5931.2022.11.001). (in Chinese)
 - [19] Fei X, Liu F, Zhang Q, Jin H, Hu H. Paving the way for NFV acceleration: A taxonomy, survey and future directions. *ACM Computing Surveys (CSUR)*, 2021, 53(4): Article No. 73. DOI: [10.1145/3397022](https://doi.org/10.1145/3397022).
 - [20] Wu M, Chen Q, Wan G J. Toward low CPU usage and efficient DPDK communication in a cluster. *The Journal of Supercomputing*, 2022, 78(2): 1852–1884. DOI: [10.1007/s11227-021-03942-x](https://doi.org/10.1007/s11227-021-03942-x).
 - [21] Belkhiri A, Pepin M, Bly M, Dagenais M. Performance analysis of DPDK-based applications through tracing. *Journal of Parallel and Distributed Computing*, 2023, 173: 1–19. DOI: [10.1016/j.jpdc.2022.10.012](https://doi.org/10.1016/j.jpdc.2022.10.012).
 - [22] Ayub A, Ishaq M, Munir M. Enhancement in multus CNI for DPDK applications in the cloud native environment. In *Proc. the 26th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, Mar. 2023, pp.66–68. DOI: [10.1109/ICIN56760.2023.10073475](https://doi.org/10.1109/ICIN56760.2023.10073475).
 - [23] Parola F, Procopio R, Querio R, Risso F. Comparing user space and in-kernel packet processing for edge data centers. *ACM SIGCOMM Computer Communication Review*, 2023, 53(1): 14–29. DOI: [10.1145/3594255](https://doi.org/10.1145/3594255).
 - [24] Miano S, Risso F, Bernal M V, Bertrone M, Lu Y. A framework for eBPF-based network functions in an era of microservices. *IEEE Trans. Network and Service Management*, 2021, 18(1): 133–151. DOI: [10.1109/TNSM.2021.3055676](https://doi.org/10.1109/TNSM.2021.3055676).
 - [25] Mostafa J, Chilingaryan S, Kopmann A. Are kernel drivers ready for accelerated packet processing using AF_XDP? In *Proc. the 2023 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Nov. 2023, pp.117–122. DOI: [10.1109/NFV-SDN59219.2023.10329590](https://doi.org/10.1109/NFV-SDN59219.2023.10329590).
 - [26] Zeng Z, Monis L, Qi S, Ramakrishnan K K. MiddleNet: A high-performance, lightweight, unified NFV and middlebox framework. In *Proc. the 8th IEEE International Conference on Network Softwarization (NetSoft)*, Jun. 27–Jul. 1, 2022, pp.180–188. DOI: [10.1109/NetSoft54395.2022.9844083](https://doi.org/10.1109/NetSoft54395.2022.9844083).
 - [27] De Benedictis M, Liroy A, Smiraglia P. Towards a secure and lightweight network function virtualisation environment. *International Journal of Grid and Utility Computing*, 2020, 11(2): 243–252. DOI: [10.1504/IJGUC.2020.105539](https://doi.org/10.1504/IJGUC.2020.105539).
 - [28] Zahoor S, Ahmad I, Rehman A U, Eldin E T, Ghamry N A, Shafiq M. Performance evaluation of virtualization methodologies to facilitate NFV deployment. *Computers, Materials and Continua*, 2023, 75(1): 311–329. DOI: [10.32604/cmc.2023.035960](https://doi.org/10.32604/cmc.2023.035960).
 - [29] Tran M N, Kim Y. Network performance benchmarking for containerized infrastructure in NFV environment. In *Proc. the 8th IEEE International Conference on Network Softwarization (NetSoft)*, Jun. 27–Jul. 1, 2022, pp.115–120. DOI: [10.1109/NetSoft54395.2022.9844100](https://doi.org/10.1109/NetSoft54395.2022.9844100).
 - [30] Nguyen D T, Dao N L, Lang K T, Pham T T, Nguyen P H, Pham C D, Pham T A, Nguyen D H, Nguyen H T. Enhancing CNF performance for 5G core network using SR-IOV in kubernetes. In *Proc. the 24th International Conference on Advanced Communication Technology (ICACT)*, Feb. 2022, pp.501–506. DOI: [10.23919/ICACT53585.2022.9728817](https://doi.org/10.23919/ICACT53585.2022.9728817).
 - [31] Tambe S D, Mandge Y, Franklin A A. Performance study of multi-access edge computing deployment in a virtualized environment. In *Proc. the 3rd IEEE 5G World Forum (5GWF)*, Sept. 2020, pp.424–429. DOI: [10.1109/5GWF49715.2020.9221113](https://doi.org/10.1109/5GWF49715.2020.9221113).
 - [32] Ramanathan S, Kondepudi K, Razo M, Tacca M, Valcarengi L, Fumagalli A. Live migration of virtual machine and container based mobile core network components: A comprehensive study. *IEEE Access*, 2021, 9: 105082–105100. DOI: [10.1109/ACCESS.2021.3099370](https://doi.org/10.1109/ACCESS.2021.3099370).
 - [33] Ventre P L, Lungaroni P, Siracusanò G, Pisa C, Schmidt F, Lombardo F, Salsano S. On the fly orchestration of unikernels: Tuning and performance evaluation of virtual infrastructure managers. *IEEE Trans. Cloud Computing*, 2021, 9(2): 710–723. DOI: [10.1109/tcc.2018.2882505](https://doi.org/10.1109/tcc.2018.2882505).
 - [34] Tu W, Wei Y H, Antichi G, Pfaff B. Revisiting the open vSwitch dataplane ten years later. In *Proc. the 2021*

- ACM SIGCOMM Conference, Aug. 2021, pp.245–257. DOI: [10.1145/3452296.3472914](https://doi.org/10.1145/3452296.3472914).
- [35] Sun C, Bi J, Zheng Z, Yu H, Hu H. NFP: Enabling network function parallelism in NFV. In *Proc. the 2017 Conference of the ACM Special Interest Group on Data Communication*, Aug. 2017, pp.43–56. DOI: [10.1145/3098822.309882](https://doi.org/10.1145/3098822.309882).
- [36] Xie S, Ma J, Zhao J. FlexChain: Bridging parallelism and placement for service function chains. *IEEE Trans. Network and Service Management*, 2021, 18(1): 195–208. DOI: [10.1109/TNSM.2020.3047834](https://doi.org/10.1109/TNSM.2020.3047834).
- [37] Zheng D, Shen G, Cao X, Mukherjee B. Towards optimal parallelism-aware service chaining and embedding. *IEEE Trans. Network and Service Management*, 2022, 19(3): 2063–2077. DOI: [10.1109/TNSM.2022.3142184](https://doi.org/10.1109/TNSM.2022.3142184).
- [38] Agarwal S, Chintapalli V R, Tamma B R. FlexSFC: Flexible resource allocation and VNF parallelism for improved SFC placement. In *Proc. the 8th IEEE International Conference on Network Softwarization (NetSoft)*, Jun. 27–Jul. 1, 2022, pp.302–306. DOI: [10.1109/NetSoft54395.2022.9844084](https://doi.org/10.1109/NetSoft54395.2022.9844084).
- [39] Chintapalli V R, Partani R, Tamma B R, Murthy C S R. Energy efficient and delay aware deployment of parallelized service function chains in NFV-based networks. *Computer Networks*, 2024, 243:110289. DOI: [10.1016/j.comnet.2024.110289](https://doi.org/10.1016/j.comnet.2024.110289).
- [40] Zhang W, Hwang J, Rajagopalan S, Ramakrishnan K K, Wood T. Flurries: Countless fine-grained NFs for flexible per-flow customization. In *Proc. the 12th International Conference on Emerging Networking EXperiments and Technologies*, Dec. 2016, pp.3–17. DOI: [10.1145/2999572.2999602](https://doi.org/10.1145/2999572.2999602).
- [41] Zeng D, Zhu A, Gu L, Li P, Chen Q, Guo M. Enabling efficient spatio-temporal GPU sharing for network function virtualization. *IEEE Trans. Computers*, 2023, 72(10): 2963–2977. DOI: [10.1109/TC.2023.3278541](https://doi.org/10.1109/TC.2023.3278541).
- [42] Kouchaksaraei H R, Venkatesh A P S, Churi A, Illian M, Karl H. Dynamic provisioning of network services on heterogeneous resources. In *Proc. the 2020 European Conference on Networks and Communications (EuCNC)*, Jun. 2020, pp.209–213. DOI: [10.1109/EuCNC48522.2020.9200912](https://doi.org/10.1109/EuCNC48522.2020.9200912).
- [43] Lopes F B, Schaeffer-Filho A E, Nazar G L. Modular VNF components acceleration with FPGA overlays. *IEEE Trans. Network and Service Management*, 2023, 20(1): 846–857. DOI: [10.1109/TNSM.2022.3211448](https://doi.org/10.1109/TNSM.2022.3211448).
- [44] Pacifico R D G, Duarte L F S, Vieira L F M, Raghavan B, Nacif J A M, Vieira M A M. eBPFlow: A hardware/software platform to seamlessly offload network functions leveraging eBPF. *IEEE/ACM Trans. Networking*, 2024, 32(2): 1319–1332. DOI: [10.1109/TNET.2023.3318251](https://doi.org/10.1109/TNET.2023.3318251).
- [45] Cziva R, Jouet S, Pezaros D P. GNFC: Towards network function cloudification. In *Proc. the 2015 IEEE Conference on Network Function Virtualization and Software Defined Network (NFV-SDN)*, Nov. 2015, pp.142–148. DOI: [10.1109/NFV-SDN.2015.7387419](https://doi.org/10.1109/NFV-SDN.2015.7387419).
- [46] Sonkoly B, Szabo R, Jocha D, Czentye J, Kind M, Westphal F J. UNIFYing cloud and carrier network resources: An architectural view. In *Proc. the 2015 IEEE Global Communications Conference (GLOBECOM)*, Dec. 2015, pp.1–7. DOI: [10.1109/GLOCOM.2015.7417869](https://doi.org/10.1109/GLOCOM.2015.7417869).
- [47] Han B, Gopalakrishnan V, Kathirvel G, Shaikh A. On the resiliency of virtual network functions. *IEEE Communications Magazine*, 2017, 55(7): 152–157. DOI: [10.1109/MCOM.2017.1601201](https://doi.org/10.1109/MCOM.2017.1601201).
- [48] Tola B, Jiang Y, Helvik B E. On the resilience of the NFV-MANO: An availability model of a cloud-native architecture. In *Proc. the 16th International Conference on the Design of Reliable Communication Networks DRCN*, Mar. 2020. DOI: [10.1109/DRCN48652.2020.1570604378](https://doi.org/10.1109/DRCN48652.2020.1570604378).
- [49] Duan J, Yi X, Zhao S, Wu C, Cui H, Le F. NFVector: A resilient NFV system using the distributed actor model. *IEEE Journal on Selected Areas in Communications*, 2019, 37(3): 586–599. DOI: [10.1109/JSAC.2019.2894287](https://doi.org/10.1109/JSAC.2019.2894287).
- [50] Liu L, Xu H, Niu Z, Li J, Zhang W, Wang P, Li J, Xue J C, Wang C. ScaleFlux: Efficient stateful scaling in NFV. *IEEE Trans. Parallel and Distributed Systems*, 2022, 33(12): 4801–4817. DOI: [10.1109/TPDS.2022.3204209](https://doi.org/10.1109/TPDS.2022.3204209).
- [51] Wu Z, Zhang Y, Feng W, Zhang Z L. NFlow and MVT abstractions for NFV scaling. In *Proc. the 2022 IEEE Conference on Computer Communications*, May 2022, pp.180–189. DOI: [10.1109/INFOCOM48880.2022.9796764](https://doi.org/10.1109/INFOCOM48880.2022.9796764).
- [52] Mimidis A, Ollora E, Soler J, Bessem S, Roulet L, Van Rossem S, Pinnerre S, Paolino M, Raho D, Du X, Chesterfield J, Flouris M, Mariani L, Riganelli O, Mobilio M, Ramos A, Labrador I, Broadbent A, Veitch P, Zembra M. The next generation platform as a service cloudifying service deployments in telco-operators infrastructure. In *Proc. the 25th International Conference on Telecommunications (ICT)*, Jun. 2018, pp.399–404. DOI: [10.1109/ICT.2018.8464838](https://doi.org/10.1109/ICT.2018.8464838).
- [53] Xu Z, Chen D, Hu Z, Sun Q. Emerging of telco cloud. *China Communications*, 2013, 10(6): 79–85. DOI: [10.1109/CC.2013.6549261](https://doi.org/10.1109/CC.2013.6549261).
- [54] Soares J, Gonçalves C, Parreira B, Tavares P, Carapinha J, Barraca J P, Aguiar R L, Sargento S. Toward a telco cloud environment for service functions. *IEEE Communications Magazine*, 2015, 53(2): 98–106. DOI: [10.1109/MCOM.2015.7045397](https://doi.org/10.1109/MCOM.2015.7045397).
- [55] Al-Jemeli M, Hussin F A. An energy efficient cross-layer network operation model for IEEE 802.15.4-based mobile wireless sensor networks. *IEEE Sensors Journal*, 2015, 15(2): 684–692. DOI: [10.1109/JSEN.2014.2352041](https://doi.org/10.1109/JSEN.2014.2352041).
- [56] Hernandez-Valencia E, Izzo S, Polonsky B. How will NFV/SDN transform service provider opex? *IEEE Network*, 2015, 29(3): 60–67. DOI: [10.1109/MNET.2015.7113227](https://doi.org/10.1109/MNET.2015.7113227).
- [57] Naik P, Shaw D K, Vutukuru M. NFVPerf: Online performance monitoring and bottleneck detection for NFV. In *Proc. the 2016 IEEE Conference on Network Function Virtualization and Software Defined Networks*

- (NFV-SDN), Nov. 2016, pp.154–160. DOI: [10.1109/NFV-SDN.2016.7919491](https://doi.org/10.1109/NFV-SDN.2016.7919491).
- [58] Liu C, Cai Z, Wang B, Tang Z, Liu J. A protocol-independent container network observability analysis system based on eBPF. In *Proc. the 26th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, Dec. 2020, pp.697–702. DOI: [10.1109/ICPADS51040.2020.00099](https://doi.org/10.1109/ICPADS51040.2020.00099).
- [59] Gardikis G, Koutras I, Mavroudis G, Costicoglou S, Xilouris G, Sakkas C, Kourtis A. An integrating framework for efficient NFV monitoring. In *Proc. the 2016 IEEE NetSoft Conference and Workshops (NetSoft)*, Jun. 2016, pp.1–5. DOI: [10.1109/NETSOFT.2016.7502431](https://doi.org/10.1109/NETSOFT.2016.7502431).
- [60] Fan J, Ye Z, Guan C, Gao X, Ren K, Qiao C. GREP: Guaranteeing reliability with enhanced protection in NFV. In *Proc. the 2015 ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization*, Aug. 2015, pp.13–18. DOI: [10.1145/2785989.2786000](https://doi.org/10.1145/2785989.2786000).
- [61] Liu J, Jiang Z, Kato N, Akashi O, Takahara A. Reliability evaluation for NFV deployment of future mobile broadband networks. *IEEE Wireless Communications*, 2016, 23(3): 90–96. DOI: [10.1109/MWC.2016.7498079](https://doi.org/10.1109/MWC.2016.7498079).
- [62] Duan Q, Wang S, Ansari N. Convergence of networking and cloud/edge computing: Status, challenges, and opportunities. *IEEE Network*, 2020, 34(6): 148–155. DOI: [10.1109/MNET.011.2000089](https://doi.org/10.1109/MNET.011.2000089).
- [63] Alshaer H. An overview of network virtualization and cloud network as a service. *International Journal of Network Management*, 2015, 25(1): 1–30. DOI: [10.1002/nem.1882](https://doi.org/10.1002/nem.1882).
- [64] Koumaras H, Tsolkas D, Garcia J, Artunedo D, Garcia B, Marco R, Salkintzis A, Fragkos D, Makropoulos G, Setaki F, Diaz A, Merino P, Koumaras V, Encinar P, Karadimas Y. A network programmability framework for vertical applications in the beyond 5G era. In *Proc. the 2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, Jun. 2022, pp.375–380. DOI: [10.1109/EuCNC/6GSummit54941.2022.9815790](https://doi.org/10.1109/EuCNC/6GSummit54941.2022.9815790).
- [65] Wang C X, You X, Gao X, Zhu X, Li Z, Zhang C, Wang H, Huang Y, Chen Y, Haas H, Thompson J S, Larsson E G, Di Renzo M, Tong W, Zhu P, Shen X, Poor H V, Hanzo L. On the road to 6G: Visions, requirements, key technologies, and testbeds. *IEEE Communications Surveys & Tutorials*, 2023, 25(2): 905–974. DOI: [10.1109/COMST.2023.3249835](https://doi.org/10.1109/COMST.2023.3249835).
- [66] Chowdhury S R, Salahuddin M A, Limam N, Boutaba R. Re-architecting NFV ecosystem with microservices: State of the art and research challenges. *IEEE Network*, 2019, 33(3): 168–176. DOI: [10.1109/MNET.2019.1800082](https://doi.org/10.1109/MNET.2019.1800082).
- [67] Hassan S, Ali N, Bahsoon R. Microservice ambients: An architectural meta-modelling approach for microservice granularity. In *Proc. the 2017 IEEE International Conference on Software Architecture (ICSA)*, Apr. 2017, pp.1–10. DOI: [10.1109/ICSA.2017.32](https://doi.org/10.1109/ICSA.2017.32).
- [68] E. Evans. Domain-Driven Design: Tackling Complexity in the Heart of Software. Addison-Wesley, 2003.
- [69] Rademacher F, Sachweh S, Zündorf A. Towards a UML profile for domain-driven design of microservice architectures. In *Proc. the 2017 Collocated Workshops: Data-Mod, FAACS, MSE, CoSim-CPS, and FOCLASA on Software Engineering and Formal Methods*, Sept. 2017, pp.230–245. DOI: [10.1007/978-3-319-74781-1_17](https://doi.org/10.1007/978-3-319-74781-1_17).
- [70] Romani Y, Tibermacine O, Tibermacine C. Towards migrating legacy software systems to microservice-based architectures: A data-centric process for microservice identification. In *Proc. the 19th IEEE International Conference on Software Architecture Companion (ICSA-C)*, Mar. 2022, pp.15–19. DOI: [10.1109/ICSA-C54293.2022.00010](https://doi.org/10.1109/ICSA-C54293.2022.00010).
- [71] Alnaim A K, Alwakeel A M, Fernandez E B. Threats against the virtual machine environment of NFV. In *Proc. the 2nd International Conference on Computer Applications & Information Security (ICCAIS)*, May 2019. DOI: [10.1109/CAIS.2019.8769561](https://doi.org/10.1109/CAIS.2019.8769561).
- [72] Kablan M, Caldwell B, Han R, Jamjoom H, Keller E. Stateless network functions. In *Proc. the 2015 ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization*, Aug. 2015, pp.49–54. DOI: [10.1145/2785989.2785993](https://doi.org/10.1145/2785989.2785993).
- [73] Kulkarni S G, Ramakrishnan K K, Wood T. Managing state for failure resiliency in network function virtualization. In *Proc. the 2020 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, Jul. 2020. DOI: [10.1109/LANMAN49260.2020.9153271](https://doi.org/10.1109/LANMAN49260.2020.9153271).
- [74] Szalay M, Nagy M, Géhberger D, Kiss Z, Mátray P, Németh F, Pongrácz G, Rétvári G, Toka L. Industrial-scale stateless network functions. In *Proc. the 12th IEEE International Conference on Cloud Computing (CLOUD)*, Jul. 2019, pp.383–390. DOI: [10.1109/CLOUD.2019.00068](https://doi.org/10.1109/CLOUD.2019.00068).
- [75] Szalay M, Mátray P, Toka L. State management for cloud-native applications. *Electronics*, 2021, 10(4): Article No. 423. DOI: [10.3390/electronics10040423](https://doi.org/10.3390/electronics10040423).
- [76] Kulkarni U, Sheoran A, Fahmy S. The cost of stateless network functions in 5G. In *Proc. the 2021 Symposium on Architectures for Networking and Communications Systems*, Dec. 2021, pp.73–79. DOI: [10.1145/3493425.3502749](https://doi.org/10.1145/3493425.3502749).
- [77] Goshi E, Karunakaran V, Harkous H, Pries R, Kellerer W. Procedure-aware stateless systems for 5G & beyond core networks. In *Proc. The 2023 IEEE Global Communications Conference*, Dec. 2023, pp.5403–5408. DOI: [10.1109/GLOBECOM54140.2023.10437306](https://doi.org/10.1109/GLOBECOM54140.2023.10437306).
- [78] Wu Y, Wang X. Research on network element management model based on cloud native technology. In *Proc. the 2nd IEEE International Conference on Computer Communication and Artificial Intelligence (CCAI)*, May 2022, pp.17–20. DOI: [10.1109/CCAI55564.2022.9807784](https://doi.org/10.1109/CCAI55564.2022.9807784).
- [79] Stolyarov R K, Shvetcova V V, Borisenko O D. TOM-MANO—Virtualised network functions management in

- cloud environment based on the TOSCA standard. *Doklady Mathematics*, 2024, 109(1): 84–92. DOI: [10.1134/S1064562424701850](https://doi.org/10.1134/S1064562424701850).
- [80] Massa J, Forti S, Paganelli F, Dazzi P, Brogi A. Declarative provisioning of virtual network function chains in intent-based networks. In *Proc. the 9th IEEE International Conference on Network Softwarization (NetSoft)*, Jun. 2023, pp.522–527. DOI: [10.1109/NetSoft57336.2023.10175449](https://doi.org/10.1109/NetSoft57336.2023.10175449).
 - [81] Mohammadi A, Nikaein N. Athena: An intelligent multi-x cloud native network operator. *IEEE Journal on Selected Areas in Communications*, 2024, 42(2): 460–472. DOI: [10.1109/JSAC.2023.3336172](https://doi.org/10.1109/JSAC.2023.3336172).
 - [82] Sharma S, Uniyal N, Tola B, Jiang Y. On monolithic and microservice deployment of network functions. In *Proc. the 2019 IEEE Conference on Network Softwarization (NetSoft)*, Jun. 2019, pp.387–395. DOI: [10.1109/NETSOFT.2019.8806705](https://doi.org/10.1109/NETSOFT.2019.8806705).
 - [83] Dab B, Fajjari I, Rohon M, Auboin C, Diquélou A. Cloud-native service function chaining for 5G based on network service mesh. In *Proc. the 2020 IEEE International Conference on Communications (ICC)*, Jun. 2020. DOI: [10.1109/ICC40277.2020.9149045](https://doi.org/10.1109/ICC40277.2020.9149045).
 - [84] Sedghpour M R S, Townend P. Service mesh and eBPF-powered microservices: A survey and future directions. In *Proc. the 2022 IEEE International Conference on Service-Oriented System Engineering (SOSE)*, Aug. 2022, pp.176–184. DOI: [10.1109/SOSE55356.2022.00027](https://doi.org/10.1109/SOSE55356.2022.00027).
 - [85] Yang W Q, Chen P F, Yu G B, Zhang H B, Zhang H X. Network shortcut in data plane of service mesh with eBPF. *Journal of Network and Computer Applications*, 2024, 222:103805. DOI: [10.1016/j.jnca.2023.103805](https://doi.org/10.1016/j.jnca.2023.103805).
 - [86] Bittar A, Wang Z, Aghasharif A, Huang C, Shami G, Lonnais M, Wilson R. Service function chaining design & implementation using network service mesh in kubernetes. In *Proc. the 7th Asian Conference on Supercomputing Frontiers*, Mar. 2022, pp.121–140. DOI: [10.1007/978-3-031-10419-0_8](https://doi.org/10.1007/978-3-031-10419-0_8).
 - [87] Duong V B, Kim Y. A design of service mesh based 5G core network using cilium. In *Proc. the 2023 International Conference on Information Networking (ICOIN)*, Jan. 2023, pp.25–28. DOI: [10.1109/ICOIN56518.2023.10049044](https://doi.org/10.1109/ICOIN56518.2023.10049044).
 - [88] Aldas S, Babakian A. Cloud-native service mesh readiness for 5G and beyond. *IEEE Access*, 2023, 11: 132286–132295. DOI: [10.1109/ACCESS.2023.3335994](https://doi.org/10.1109/ACCESS.2023.3335994).
 - [89] Osmani L, Kauppinen T, Komu M, Tarkoma S. Multi-cloud connectivity for kubernetes in 5G networks. *IEEE Communications Magazine*, 2021, 59(10): 42–47. DOI: [10.1109/MCOM.110.2100124](https://doi.org/10.1109/MCOM.110.2100124).
 - [90] Aditya P, Akkus I E, Beck A, Chen R C, Hilt V, Rimac I, Satzke K, Stein M. Will serverless computing revolutionize NFV? *Proceedings of the IEEE*, 2019, 107(4): 667–678. DOI: [10.1109/JPROC.2019.2898101](https://doi.org/10.1109/JPROC.2019.2898101).
 - [91] Savi M, Banfi A, Tundo A, Ciavotta M. Serverless computing for NFV: Is it worth it? A performance comparison analysis. In *Proc. the 2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, Mar. 2022, pp.680–685. DOI: [10.1109/PerComWorkshops53856.2022.9767495](https://doi.org/10.1109/PerComWorkshops53856.2022.9767495).
 - [92] Shen J, Yu H, Zheng Z, Sun C, Xu M, Wang J. Serpens: A high-performance serverless platform for NFV. In *Proc. the 28th IEEE/ACM International Symposium on Quality of Service (IWQoS)*, Jun. 2020. DOI: [10.1109/IWQoS49365.2020.9213030](https://doi.org/10.1109/IWQoS49365.2020.9213030).
 - [93] Femminella M, Reali G. Implementing Internet of Things service platforms with network function virtualization serverless technologies. *Future Internet*, 2024, 16(3): Article No. 91. DOI: [10.3390/fi16030091](https://doi.org/10.3390/fi16030091).
 - [94] Suzuki T, Kim S Y, Kani J I, Yoshida T. Real-time polarization demultiplexing by multi-thread constant modulus algorithm for fully softwarized access networks. *Journal of Lightwave Technology*, 2023, 41(5): 1346–1356. DOI: [10.1109/JLT.2022.3225061](https://doi.org/10.1109/JLT.2022.3225061).
 - [95] Almeida L F, Pereira S S, Domingues J D, Oliveira A S R, Carvalho N B. Moving NFV toward the antenna through FPGA-based hardware reconfiguration. *IEEE Communications Letters*, 2023, 27(1): 342–346. DOI: [10.1109/LCOMM.2022.3216506](https://doi.org/10.1109/LCOMM.2022.3216506).
 - [96] Cong J, Fang Z, Huang M, Wei P, Wu D, Yu C H. Customizable computing—From single chip to datacenters. *Proceedings of the IEEE*, 2019, 107(1): 185–203. DOI: [10.1109/JPROC.2018.2876372](https://doi.org/10.1109/JPROC.2018.2876372).
 - [97] Lal R, Anderson J B, Jackson A. Data processing unit's entry into confidential computing. In *Proc. the 12th International Workshop on Hardware and Architectural Support for Security and Privacy*, Oct. 2023, pp.56–63. DOI: [10.1145/3623652.3623670](https://doi.org/10.1145/3623652.3623670).
 - [98] Li B, Hou F, Yang G, Zhao H, Chen S. Data analysis-oriented stochastic scheduling for cost efficient resource allocation in NFV based MEC network. *IEEE Trans. Vehicular Technology*, 2023, 72(5): 6695–6708. DOI: [10.1109/TVT.2023.3234285](https://doi.org/10.1109/TVT.2023.3234285).
 - [99] Cisneros J C, Yanguis S, Hernández S E P, Drira K. A survey on distributed NFV multi-domain orchestration from an algorithmic functional perspective. *IEEE Communications Magazine*, 2022, 60(8): 60–65. DOI: [10.1109/MCOM.002.2100950](https://doi.org/10.1109/MCOM.002.2100950).
 - [100] Abu-Lebdeh M, Naboulsi D, Glitho R, Tchouati C W. On the placement of VNF managers in large-scale and distributed NFV systems. *IEEE Trans. Network and Service Management*, 2017, 14(4): 875–889. DOI: [10.1109/TNSM.2017.2730199](https://doi.org/10.1109/TNSM.2017.2730199).
 - [101] Shen X, Gao J, Wu W, Li M, Zhou C, Zhuang W. Holistic network virtualization and pervasive network intelligence for 6G. *IEEE Communications Surveys & Tutorials*, 2022, 24(1): 1–30. DOI: [10.1109/COMST.2021.3135829](https://doi.org/10.1109/COMST.2021.3135829).
 - [102] Mehmood K, Kravlevska K, Palma D. Intent-driven autonomous network and service management in future cellular networks: A structured literature review. *Computer Networks*, 2023, 220:109477. DOI: [10.1016/j.comnet.2022.109477](https://doi.org/10.1016/j.comnet.2022.109477).
 - [103] Huang Y X, Chou J. A survey of NFV network acceleration

tion from ETSI perspective. *Electronics*, 2022, 11(9): 1457. DOI: [10.3390/electronics11091457](https://doi.org/10.3390/electronics11091457).

- [104] Zhang T Z, Qiu H, Linguaglossa L, Cerroni W, Giaccone P. NFV platforms: Taxonomy, design choices and future challenges. *IEEE Trans. Network and Service Management*, 2021, 18(1): 30–48. DOI: [10.1109/TNSM.2020.3045381](https://doi.org/10.1109/TNSM.2020.3045381).
- [105] Souza R, Dias K, Fernandes S. NFV data centers: A systematic review. *IEEE Access*, 2020, 8: 51713–51735. DOI: [10.1109/ACCESS.2020.2973568](https://doi.org/10.1109/ACCESS.2020.2973568).
- [106] Zoure M, Ahmed T, Reveillère L. Network services anomalies in NFV: Survey, taxonomy, and verification methods. *IEEE Trans. Network and Service Management*, 2022, 19(2): 1567–1584. DOI: [10.1109/TNSM.2022.3144582](https://doi.org/10.1109/TNSM.2022.3144582).
- [107] Saraiva De Sousa N F, Lachos Perez D A, Rosa R V, Santos M A S, Esteve Rothenberg C. Network service orchestration: A survey. *Computer Communications*, 2019, 142/143: 69–94. DOI: [10.1016/j.comcom.2019.04.008](https://doi.org/10.1016/j.comcom.2019.04.008).
- [108] Kaur K, Mangat V, Kumar K. A comprehensive survey of service function chain provisioning approaches in SDN and NFV architecture. *Computer Science Review*, 2020, 38:100298. DOI: [10.1016/j.cosrev.2020.100298](https://doi.org/10.1016/j.cosrev.2020.100298).
- [109] Santos G L, de Freitas Bezerra D, da Silva Rocha É, Ferreira L, Moreira A L C, Gonçalves G E, Marquezini M V, Recse Á, Mehta A, Kelner J, Sadok D, Endo P T. Service function chain placement in distributed scenarios: A systematic review. *Journal of Network and Systems Management*, 2022, 30(1): Article No. 4. DOI: [10.1007/s10922-021-09626-4](https://doi.org/10.1007/s10922-021-09626-4).
- [110] Azadiabad S, Khendek F. Dependability of network services in the context of NFV: A taxonomy and state of the art classification. *Journal of Network and Systems Management*, 2024, 32(2): Article No. 37. DOI: [10.1007/s10922-024-09810-2](https://doi.org/10.1007/s10922-024-09810-2).



Shuo Quan received his Bachelor's and Master's degrees in software engineering from Beijing University of Posts and Telecommunications, Beijing, in 2013 and 2016, respectively. He is a researcher at China Telecom

Cloud Computing Research Institute, Beijing. His research interests include cloud computing, networking and communication systems, cloud network convergence, cloud-native and virtualization, etc.



Shen Gao received his Bachelor's degree in computer science and technology from Tianjin University, Tianjin, in 2016, and his Master's degree in computer science from the Illinois Institute of Technology, Illinois, in 2018. He is currently employed at the Chi-

na Telecom Cloud Computing Research Institute, Beijing. His research interests include NFV, cloud computing, and 5G.



Jian-Song Zhang received his Bachelor's and Master's degrees in automation from Tsinghua University, Beijing, in 2004 and 2006, respectively, and his Ph.D. degree in computer science and engineering from the Hong Kong University of Science and Tech-

nology, Hong Kong, in 2014. Currently, he is a senior researcher in China Telecom. His research interests include cloud computing, networking and communication systems, computer architecture and heterogeneous computing, ubiquitous computing, and vehicle computing, etc.



Jie Wu is Laura H. Carnell Professor at Temple University and the Director of the Center for Networked Computing (CNC), Temple University, Philadelphia. His current research interests include mobile computing and wireless networks, routing proto-

cols, network trust and security, distributed algorithms, applied machine learning, and cloud computing. Currently, Dr. Wu is on leave working at China Telecom as a scientist in cloud computing.