# Towards low-latency service delivery in a continuum of virtual resources: State-of-the-art and Research Directions

José Santos, *Student Member, IEEE,* Tim Wauters, *Member, IEEE,* Bruno Volckaert, *Member, IEEE,* and Filip De Turck, *Fellow, IEEE*

***Abstract*—The advent of softwarized networks has enabled the deployment of chains of virtual network and service components on computational resources from the cloud up to the edge, creating a continuum of virtual resources. The next generation of low latency applications (e.g. Virtual Reality (VR), autonomous cars) adds even more stringent requirements to the infrastructure, calling for considerable advancements towards cloud-native micro-service-based architectures. This article presents a comprehensive survey on ongoing research aiming to effectively support low latency services throughout their execution lifetime in next-generation networks. The current state-of-the-art is critically reviewed to identify the most promising trends that will strongly impact the full applicability and high performance of low latency services. This article proposes a taxonomy as well as specific evaluation criteria to classify research across different domains addressing low latency service delivery. Current architectural paradigms such as Multi-access Edge Computing (MEC) and Fog Computing (FC) alongside novel trends on communication networks are discussed. Among these, the integration of Machine Learning (ML) and Artificial intelligence (AI) is introduced as a key research field in current literature towards autonomous network management. A discussion on open challenges and future research directions on low-latency service delivery leads to the conclusion, offering lessons learned and prospects on emerging use cases such as Extended Reality (XR), in which novel trends will play a major role.**

***Index Terms*—Low latency, Next-generation networks, Cloud-native, autonomous networks, orchestration, 5G/6G**

## I. Introduction

The deployment of high-bandwidth and low latency 5G network infrastructures has been driving the digital transformation of network services in Industry 4.0, Smart Cities, Healthcare and connected vehicles. Founded on the principles of Software-Defined Networking (SDN) [1] and Network Function Virtualization (NFV) [2], programmable networks interconnect virtual cloud, fog, and edge resources, which help to bring low latency services to reality. These technological advancements pave the way to support high reliability and low latency services in 5G such as Ultra-Reliable Low-Latency Communication (URLLC) services required for autonomous driving and factory automation use cases. The massive growth of the Internet of Things (IoT) is pushing the boundaries of network architectures by transforming everyday objects into smart connected devices. To overcome the hurdles to arrive at truly end-to-end (E2E) services, which meet the even more stringent requirements (e.g. higher bandwidths, lower latencies) of future applications, next-generation (6G) networks [3] have to provide distributed orchestration and management functionalities to integrate a continuum of virtual computing resources with a wide variety of ultra-broadband (radio access and core) and high-precision network links. Bandwidth requirements for Extended Reality (XR) or Holographic Type Communication (HTC) applications will rise well above 1Tbps, while their interactive experiences require sub-millisecond latencies [4]. Pervasive and ambient connectivity (billions of devices per km$^2$ in the Internet of Everything (IoE) with ultra-low latency, near-proximity communication) and autonomous service delivery (with levels up to 99.99999% reliability, even at high speeds of users or Unmanned Aerial Vehicles (UAVs) [5]) add further stringent requirements, as shown in Table I. Supported by pervasive network telemetry and analytics, Artificial Intelligence (AI) capable of dynamically meeting the real-time requirements of cloud-native applications will be brought into play as well. Distributed learning at both the network and application levels will contribute to the development of smart, highly interactive and reliable environments, allowing for a high Quality of Experience (QoE) to end-users. On the business side, increasingly more stakeholders will be involved in the E2E service chain, requiring very dynamic service contracts and relationships beyond mere network connectivity, expanding towards control and management aspects. Extensions to the ETSI NFV MANO model [6] should incorporate these micro-operators, their interfaces, roles and templates to accelerate network slice setup.

To ensure low E2E service latency for all emerging use cases, current network architectures need to drastically change. Several improvements are currently being implemented at the Radio Access Network (RAN) and core alongside novel networking systems incorporating SDN, NFV and caching concepts. Multiple steps in the service execution causing increased delays have to be recognized as potential barriers to low latency service delivery. Distributed (hierarchical) SDN architectures are considered better candidates than current centralized approaches [7]. However, further research is needed to understand how such architectures can incorporate local or global network measurements and analysis to steer the message routing, and how service-level objectives can be enforced

Authors are with Ghent University - imec, IDLab, Department of Information Technology, Belgium. Email: josepedro.pereiradossantos@UGent.be

Manuscript received April 28, 2021

TABLE I: Requirements for emerging use cases [3], [4].

| Use Case | Latency | Reliability | Throughput |
|---|---|---|---|
| E-Health | < 1 ms | 99.99999% | 1 - 100 Mbps |
| Smart Cities | 10ms - 1s | > 99.999% | 1 - 100 Mbps |
| UAV services | 1 - 10 ms | 99.99999% | 1 - 10 Mbps |
| Industrial IoT | 1 - 10 ms | 99.99999% | 1 - 10 Mbps |
| Extended Reality | < 1 ms | 99.99999% | > 1 Tbps |
| Self-driving cars | < 1 ms | 99.99999% | 1 - 10 Mbps |

in the network. In the network path, fog-cloud infrastructures need to be set up to execute several micro-services of a service chain, enabling flexible deployments [8]. Moreover, the integration of intelligence at the edge will lead to ML-driven networks able to support highly dynamic management updates under varying network circumstances, meeting the requirements of cloud-native applications and services over the continuum of virtual resources. This article revisits several aspects of the state-of-the-art on low latency service delivery in next-generation networks. The contributions of the article can be summarized as follows:

- Present up-to-date research and novel trends in low latency service delivery by conducting a comprehensive review of the current literature.
- Propose a taxonomy on low latency service delivery by considering different aspects of next-generation networks that will impact the applicability and performance of low latency services.
- Provide specific evaluation criteria to classify research across different domains based on the presented taxonomy.
- Identify current open challenges and future directions.
- Provide lessons learned and prospects of emerging use cases such as Smart Cities and XR.

Despite the importance of low latency services in next-generation networks, to the best of our knowledge, a comprehensive and detailed survey on novel trends for low latency service delivery is still missing. The close interplay between computing and networking is key to support low latency 6G services in the future. The integration of AI/ML at the edge will also play a major role in enabling autonomous networks. This article presents valuable insights for the industry and research community into ongoing research and novel trends pushing towards distributed cloud-native infrastructures capable of supporting low E2E service latency. The next section outlines how our survey differs from the current state-of-the-art and presents the taxonomy on low latency service delivery.

## II. Survey Methodology

This section starts by introducing the existing surveys and tutorials in the literature related to low latency services. Second, our literature review is explained followed by a taxonomy on low latency service delivery. Then, the evaluation criteria used to classify research across different domains is introduced. Finally, the article structure is presented.

### A. Existing Surveys & Tutorials

Several surveys and tutorials on 5G networks are available in the literature. In [9], architectural paradigms and emerging technologies are presented. 5G research projects are also briefly introduced. In [10] and [11], SDN and NFV concepts for 5G networks are addressed. Both surveys highlight how SDN and NFV complement each other and discuss the key role both technologies will play on next-generation networks. In [12], the integration of Multi-access Edge Computing (MEC) in 5G systems is assessed. It highlights the deployment of applications and services at the edge as one of the main benefits of MEC. Service migration and mobility support in MEC are also addressed in [13]. In [14], RAN, core network and caching concepts for 5G are discussed in detail, while in [15] resource management for 5G RAN systems is considered. Recently, in [16], service placement in Fog Computing (FC) is discussed while a comprehensive taxonomy for FC is proposed in [17]. The classification of FC applications based on ML is also presented in [18]. Hardware-accelerated platforms and infrastructures (e.g. Field-Programmable Gate Arrays (FPGAs), microprocessors) are also covered in [19], highlighting relevant studies for the softwarized execution of network services. Furthermore, a few surveys have already been published on 6G networks focused on architectures [20], [21] and wireless access networks [22]. Regarding applications, surveys and tutorials on emerging use cases exist such as Smart Cities [23], Autonomous cars [24], Augmented Reality (AR) [25] and Industrial IoT (IIoT) [26].

Low latency has also been addressed in recent surveys. Previous work [27] discusses latency reduction techniques by comparing their advantages with their overhead in implementation and deployment. The authors focus on communication protocols and how these techniques impact the latency perceived by end-users. Novel architectures and emerging applications fall out of their scope. Design principles and enabling technologies to deploy low-latency wireless communication networks have been analyzed in [28]. The authors reflect on how to meet the stringent requirements of future use cases while discussing the trade-offs between low latency and traditional performance metrics. Other surveys related to low latency exist in the current literature, but their scope is limited [29] or focused on a specific use case [30]. These surveys do not assess each contribution in the presence of specific criteria, as it is performed in this article. An exhaustive literature review is also missing, especially considering novel trends on low latency service delivery. In contrast, this article comprehensively reviews recent research and novel trends focused on enabling services that require low E2E latency in next-generation networks while proposing a taxonomy on low latency service delivery.

### B. Literature Review

The literature on low latency service delivery encompasses several domains, thus structuring and classifying the most relevant research is not a trivial task. Fig. 1 shows the proposed taxonomy. Based on an exhaustive literature search, six main categories have been identified: *Architecture*, *Network*,

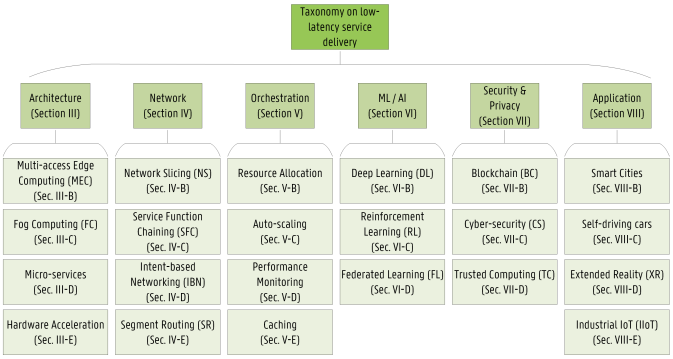



Fig. 1: Overview of the surveyed research.

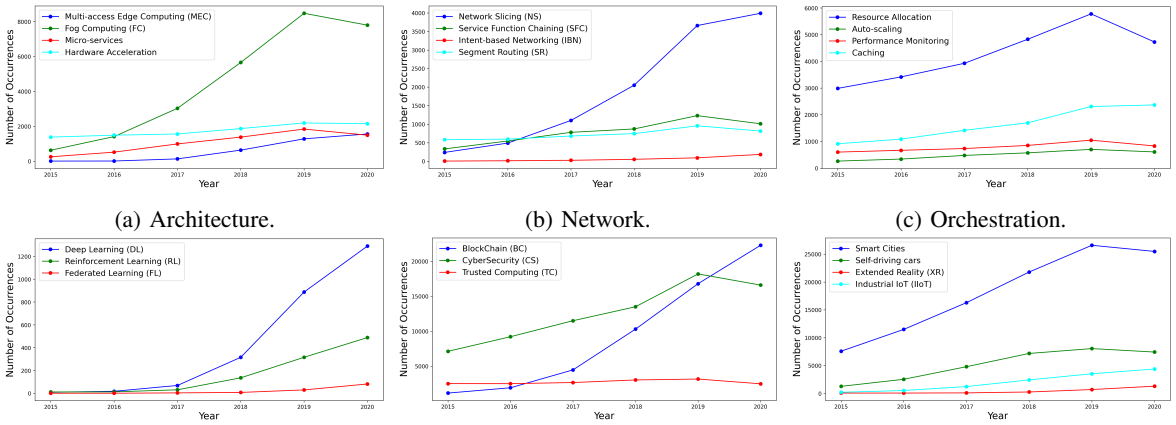
*Orchestration*, *Security & Privacy*, *ML / AI* and *Application*. These six domains have been identified as crucial for the full applicability and high performance of emerging low latency services in next-generation networks. The review of research contributions on all six domains allows us to identify open issues relevant for several areas and aggregate reviewed work under these main categories. Efforts on radio and wireless access networks have been excluded from our survey since these topics are usually covered in dedicated surveys [31], [32]. Nevertheless, the importance of improving current access networks is acknowledged. Efforts focused only on SDN and NFV paradigms have also been excluded since these topics have been thoroughly addressed in the literature [33].

After performing various reiterative fine-tuned search processes based on multiple keywords, several research domains have been recognized under the six main categories. By dividing the main category into different domains, readers can easily access references addressing a specific issue. Within the first main category, *Architecture*, four research domains have been identified: MEC, FC, Micro-services and Hardware acceleration. These domains have been perceived as important enablers towards fully cloud-native infrastructures in next-generation networks. Research related to Cloudlets [34] has been left out since both MEC and FC are emerging in the last few years and are the main alternatives in this domain. Regarding the second category, *Network*, four research domains have been distinguished: Network Slicing (NS), Service Function Chaining (SFC), Intent-based Networking (IBN) and Segment Routing (SR). Within the third main category, *Orchestration*, four domains have been identified: Resource allocation, Auto-scaling, Performance monitoring and Caching. These areas demonstrate the importance of efficient management practices in the life-cycle of service components. The fourth category, *Security & Privacy*, addresses three research domains: BlockChain (BC), CyberSecurity (CS) and Trusted Computing (TC). Recent security guidelines focused on data protection, trust and authentication/authorization mechanisms have been identified since security aspects are commonly left out in current literature. This article tackles this gap in the state-of-the-art by reviewing novel security trends. Three domains have been recognized in the fifth main category, *ML / AI*: Deep Learning (DL), Reinforcement Learning (RL) and Federated Learning (FL). These emerging areas in ML/AI research are being investigated in the context of distributed clouds towards autonomous management. Finally, within the sixth category, *Application*, four research areas have been identified: Smart Cities, Self-driving cars, XR and IIoT. These application domains represent emerging use cases in next-generation networks, where low latency is crucial for the proper service operation and end-user satisfaction.

Concerning the literature review, keywords based on the presented taxonomy have been searched on two publication databases, Google Scholar and IEEE Xplore. The number of occurrences in Google Scholar for all keywords is shown in Fig. 2. For example, the search term used to determine the number of occurrences of FC has been *Architecture "Fog Computing"*. Similar keywords have been used for the other research domains in our taxonomy. Some domains were still unknown in 2017 until a significant increase occurred in 2019, especially regarding IBN and DL research. Some topics are still largely unexplored in academia as FL and IBN, while BC and DL had a tremendous increase over the last three years. Based on these graphs, research published between January 2017 and December 2020 has been examined. Table II shows the number of publications reviewed in this article per main category and correspondent domain. Publications have been organized per chronological order on each domain subsection to improve readability since several works fall under different criteria. Moreover, works incorporate concepts falling in different categories, thus, the corresponding percentages are derived accordingly. For instance, for Architectural paradigms, eight papers (15.8%) have been analyzed in the context of MEC, six papers for FC (13.3%) while four papers have been reviewed for both Micro-services (5.8%) and Hardware acceleration (4.1%). The inclusion of research is based on the overall quality of the publication (i.e. number of citations, journal indexed in Web of Science, etc). Only peer-reviewed works have been considered and short conference papers (typically between 2-4 pages) have not been included. Based on these principles, 120 publications have been selected to be thoroughly analyzed in the context of this article. The next subsection describes in detail the evaluation criteria applied to classify research on low latency service delivery.

### C. Evaluation Criteria

The criteria to evaluate current research on low latency service delivery for next-generation networks are presented in Table III. Although low latency is the main focus, other requirements are also important to ensure low E2E latency. Several requirements have been translated into individual criteria. If the work addresses any of the given criteria in their methodology (e.g. architecture, algorithm), then it meets the individual criterion. Otherwise, the criterion is unmet when this requirement is not considered. The first criterion (C1) is supporting *Mobility*. Devices and end-users will request services on-demand at different locations. Without efficient mobility support, service discovery procedures for mobile devices may need to restart, causing service disruptions and degraded user experience. Mobility is crucial for low latency services to guarantee service continuity when end-users or




(a) Architecture. (b) Network. (c) Orchestration.

(d) ML/AI. (e) Security & Privacy (f) Application.

Fig. 2: Search occurrence of keywords based on the proposed taxonomy on low latency service delivery in Google Scholar.

TABLE II: The number of publications assessed per category and correspondent research domains.

| Main Category | Research Domain | Number of Publications |
|---|---|---|
| Architecture (39.0%) | Multi-access Edge Computing | 8 (15.8%) |
| | Fog Computing | 6 (13.3%) |
| | Micro-services | 4 (5.8%) |
| | Hardware Acceleration | 4 (4.1%) |
| Network (30.7%) | Network Slicing | 4 (8.3%) |
| | Service Function Chaining | 7 (13.3%) |
| | Intent-based Networking | 5 (4.1%) |
| | Segment Routing | 6 (5.0%) |
| Orchestration (27.3%) | Resource Allocation | 8 (13.3%) |
| | Auto-scaling | 4 (3.3%) |
| | Performance Monitoring | 5 (4.1%) |
| | Caching | 6 (6.6%) |
| ML / AI (19.9%) | Deep Learning | 5 (7.5%) |
| | Reinforcement Learning | 6 (8.3%) |
| | Federated Learning | 5 (4.1%) |
| Security & Privacy (27.4%) | Blockchain | 6 (10.0%) |
| | Cybersecurity | 5 (14.1%) |
| | Trusted Computing | 4 (3.3%) |
| Application (24.8%) | Smart Cities | 7 (10.8%) |
| | Self-driving cars | 4 (4.1%) |
| | Extended Reality | 4 (3.3%) |
| | Industrial IoT | 7 (6.6%) |

devices are moving in the network, ensuring smooth handover processes. The second criterion (C2) is *Scalability*. Cloud-native infrastructures need to support millions of connected devices for multiple applications. The network demand at a given time determines the computing resources allocated to each application, meaning that these systems need to be elastic enough to scale up and down resources according to the current demand. The third criterion (C3) relates to *Energy Efficiency*. Latency and energy efficiency are often considered opposing strategies. If service providers offer low E2E latency, that usually translates into higher energy bills since more computing resources are allocated to provide lower latency to all users (limited resource sharing). Users want the best Quality of Service (QoS) for the minimum cost, while service providers want to meet the agreed QoS level by using a small fraction of their infrastructure, reducing their operational costs and maximizing their profit. The massive number of connected devices will generate a huge volume of data that, if processed centrally by traditional clouds, increases power consumption and latency. Efficient collaboration between edge, fog and cloud is crucial towards a more efficient and greener cloud-native infrastructure [35]. The fourth criterion (C4) reflects on *Isolation*. In next-generation networks, resources should be logically separated through slicing, an abstraction allowing resource sharing among several slices while preventing attacks and faults on a slice from affecting any other slices in the network. A network slice can be defined as a set of Network Functions (NFs) and resources (e.g. connectivity, storage) attributed to a specific service. These functions are chained to form a logical isolated E2E network slice responsible for offering the specified QoS level. NS is expected to significantly reduce operational costs and guarantee different latency demands for each supported application.

The fifth criterion (C5) relates to efficient *Security and Privacy* mechanisms. Security mainly deals with the integrity and availability of the system, while privacy concerns data protection and confidentiality. Any security breach can directly impact the privacy of the company or the individual if unauthorized access is granted to private data. In the past, security practices protected enterprises through perimeter-based solutions. Nowadays, data is spread across the network and stored at different levels (i.e. edge, fog, cloud), making traditional security practices inadequate. Security will be key to most emerging use cases since users would only subscribe to services where their privacy and data are protected. Thus, identifying security and privacy issues in distributed clouds will be the main challenge in security research for the next coming years. The sixth criterion (C6) concerns *Resilience*. Network resiliency is the ability to self-heal the network after unexpected failures [36]. Next-generation networks should offer self-healing features to yield sustained QoS levels under critical situations by predicting or identifying network failures. Reactive and proactive remedies can ensure service continuity and availability. The seventh criterion (C7) is *Reliability*. Reliability is often compared to availability, but a service can be available and not run properly. High reliability will

TABLE III: The evaluation criteria applied to classify research on low latency service delivery.

| Evaluation criteria | Description |
|---|---|
| Mobility (C1) | Mobility is an important requirement for next-generation networks where services can be requested on-demand by multiple devices at different locations. |
| Scalability (C2) | Millions of devices will be connected to the network. Cloud-native infrastructures need to accommodate applications with different latency requirements while adapting to the current network demand. |
| Energy Efficiency (C3) | The number of connected devices and the data they collect is growing exponentially. If data is processed centrally by traditional clouds, it would increase power consumption and latency. Cooperation between the edge, fog and the cloud will become even more important in future networks given the increased use of distributed ML techniques and data processing. |
| Isolation (C4) | Network slicing enables the efficient execution of different services on the same infrastructure with various latency thresholds. |
| Security & Privacy (C5) | Traditional security solutions are designed to protect enterprise networks and data centers through perimeter-based protections. These practices are no longer adequate for addressing security challenges in emerging use cases. For example, distributed malware monitoring observes low-powered devices to compensate for their limited security. |
| Resilience (C6) | Resilience must be an inherent property into next-generation networks. If failures occur, networks need to keep offering satisfactory Quality of Service (QoS) no matter what challenges they face. Otherwise, time and money are lost. |
| Reliability (C7) | Reliability is essential for low latency services in next-generation networks and will be even harder to maintain in distributed cloud-native infrastructures. Devices and end-users can connect from multiple locations with different access technologies, causing failures at the edge, fog, or cloud. |
| Heterogeneity (C8) | Devices hold various computing capacities (e.g. CPU, RAM) and will access the medium through different technologies (e.g. 5G, Wi-Fi). Cloud-native infrastructures must ensure such heterogeneous networks run smoothly. |
| Throughput (C9) | Future networks need to support low latency service delivery as well as high bandwidth data rates. XR is among the most throughput demanding emerging applications. |
| Federation (C10) | Computing resources will be geographically distributed across the edge, fog and cloud. Several domains can be operated via different service providers. Federation is required to manage and orchestrate services running on different providers that compose a complete application. Low E2E latency must be guaranteed under cloud-native federation. |

be crucial for low latency services since service providers are expected to match the agreed QoS levels to satisfy their users. Reliability is also associated with resilience. Reliability is the goal of service providers, while resiliency contributes to its accomplishment. Services can be reliable since no failures have occurred, but they cannot be considered resilient since such failure-tolerant capabilities have not been tested. The eighth criterion (C8) relates to network *Heterogeneity*. Infrastructure nodes will hold different computing capabilities. For example, edge and fog nodes possess limited capacity compared to cloud nodes. This heterogeneity needs to be considered in resource allocation decisions - where and when to deploy service components is essential to manage the network heterogeneity and ensure services run as expected. The ninth criterion (C9) is *Throughput*. Emerging use cases such as XR require not only low latency but also high bandwidth data rates. Several network functionalities (e.g. data processing, ML operations) are currently being pushed to the edge to minimize latency and maximize throughput. Finally, the tenth criterion (C10) is supporting *Federation*. Cloud-native infrastructures will be operated via different service providers. Applications following the recent micro-service paradigm will be separated into multiple service components that can be deployed over distinct providers in a continuum of virtual resources from the cloud up to the edge. Cooperation over various network domains ensures proper management and orchestration of these applications and delivers low E2E latency.

### D. Article Structure

The remainder of the article is organized as follows: Section III presents architectural paradigms such as MEC and FC and respective literature. Section IV introduces and reviews recent progress in communication networks. Section V reviews novel management and orchestration practices followed by recent advances on ML and AI towards automated management in section VI. Section VII introduces novel security and privacy methods for next-generation networks. Section VIII reviews the most prominent low latency use cases. Section IX focuses on open challenges and future directions while Section X presents the lessons learned and discusses the prospects of emerging use cases such as Smart Cities and XR, in which novel trends will play a key role. Concluding remarks are presented in Section XI.

## III. Towards a continuum of virtual resources - Architectural paradigms

### A. Overview

The advent of novel architectural paradigms enabled the deployment of service chains on computational resources from the cloud up to the edge. This brings several benefits such as low latency and mobility support. This section presents the key architectural concepts enabling application deployment in a continuum of virtual resources: MEC, FC, micro-services and hardware acceleration. Then, related research is discussed, in which the works are categorized based on our criteria. Table IV summarizes the reviewed works.

### B. Multi-Access Edge Computing (MEC)

MEC [59] is an industry initiative from the European Telecommunication Standards Institute (ETSI). It was launched in 2014 under a different naming: Mobile Edge Computing focused on bringing the current mobile network to the edge by adding Virtual Machine (VM) virtualization. In 2017, ETSI incorporated non-cellular operators' requirements (e.g. MEC hosts deployed in multiple networks owned by different providers, edge applications running collaboratively), thus the name changed to Multi-Access Edge Computing (MEC). The ETSI MEC technical committee is designing a reference architecture [60] for a mobile edge system as shown

TABLE IV: Summary of the reviewed works in terms of Architecture.

| Research Domain | Authors | Main focus | Year | Evaluation Criteria | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | C10 |
| Multi-access Edge Computing (MEC) (Sec. III-B) | Liu, X., et al. [37] | Service migration | 2017 | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Ma, L., et al. [38] | Mobility support | 2018 | ✓ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Hsieh, H., et al. [39] | Edge Intelligence | 2018 | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ |
| | Liu, J., et al. [40] | Network interoperability | 2018 | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ |
| | Yang, S., et al. [41] | Video streaming | 2018 | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ |
| | Shah, S., et al. [42] | Mobility Support | 2020 | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✓ |
| | Ranaweera, P., et al. [43] | Security | 2020 | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Ksentini, A., et al. [44] | Slicing in MEC | 2020 | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Fog Computing (FC) (Sec. III-C) | Sookhak, M., et al. [45] | Vehicular networks | 2017 | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| | Moreno-V., R., et al. [46] | FC architecture | 2017 | ✓ | ✓ | ✗ | ✗ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ |
| | Sharma, P. K., et al. [47] | SDN-based FC | 2017 | ✗ | ✓ | ✗ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ |
| | Bruschi, R., et al. [48] | SDN-based FC | 2017 | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Baccarelli, E., et al. [49] | IoT services | 2017 | ✓ | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ |
| | Santos, J., et al. [50] | Smart Cities | 2018 | ✓ | ✗ | ✓ | ✗ | ✓ | ✗ | ✓ | ✓ | ✗ | ✗ |
| Micro-services (Sec. III-D) | Brenner, S., et al. [51] | Security | 2017 | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Guija, D., et al. [52] | Security | 2018 | ✗ | ✓ | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ |
| | Xu, R., et al. [53] | Blockchain architecture | 2019 | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Debauche, O., et al. [54] | Edge architecture | 2020 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| Hardware Acceleration (Sec. III-E) | Zhang, X., et al. [55] | scalable NFV | 2017 | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| | Umuroglu, Y., et al. [56] | Neural networks | 2017 | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| | Cai, R., et al. [57] | Neural networks | 2018 | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Owaida, M., et al. [58] | FPGA-based acceleration | 2019 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |

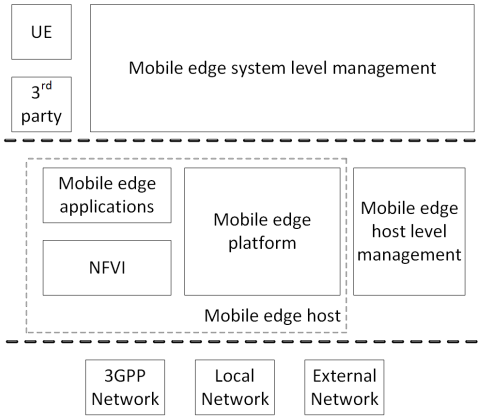



Fig. 3: The mobile edge system reference architecture [50].

in Fig. 3. MEC focuses on evolving the mobile network edges to create a cloud environment close to the RAN that hosts enhanced services provided by the Mobile Network Operator (MNO) or third parties. Mobile Edge (ME) applications run on top of a generic cloud infrastructure within the RAN: the Mobile Edge Host (MEH). The MEH encompasses a Mobile Edge Platform (MEP) responsible for executing ME applications on an NFV Infrastructure (NFVI), which provides computing, storage and network resources for the provisioning and consumption of ME services [60]. The envisioned use cases include IoT, AR, optimized caching and video analytics.

MEC aims to improve the mobile architecture to support low latency services, but a few challenges persist: services may need to be reallocated quickly, thus service migration and mobility support present an enormous challenge. In [37], Liu, X., et al. have proposed a mobility-aware coded probabilistic caching scheme for MEC-enabled small cells. The authors aim to maximize throughput by incorporating user mobility (C1) and distributed storage in their scheme. Results have shown that their scheme outperforms conventional probabilistic methods, attaining higher throughput (C9) under different degrees of user mobility, content popularity and backhaul capability. In [38], Ma, L., et al. have presented an edge architecture supporting seamless migration (C1) of offloading services while keeping moving users connected to their nearest edge server. Docker container migration has been recognized as an important challenge in the literature. The authors propose to leverage the layered nature of the Docker storage system to reduce file system synchronization overhead. Their framework provides an isolated (C4) running environment for the offloading service via two layers of the system virtualization hierarchy. It also minimizes security risks (C5) posed to offloading services running on the edge servers. The authors state that isolation between different services provides a degree of security. The authors have evaluated the performance of container migration based on several metrics, such as latency, file compression and throughput (C9). Both works are promising, however, no clear strategy or guidelines have yet been defined on how to support mobility in future networks.

Recent works have also focused on providing intelligence at the edge and in the interoperability between wireless and wired networks or different networking components. In [39], Hsieh, H. et al. have studied virtualized MEC (vMEC) infrastructures to provide intelligence at the edge while reducing latency and increasing the available capacity. Their vMEC infrastructure applies container-based virtualization as an IoT gateway for flow control mechanisms and performance analysis. Results demonstrate that latency can be reduced up to 30%, maintaining high bandwidth for most services. The flow control mechanism has been introduced to reduce the CPU usage of the vMEC platform, reducing energy consumption (C3). Their approach not only reduces latency but also enhances user experience by improving service quality, which relates to

reliability (C7). The authors have also focused on adjusting the throughput capacity with their flow control mechanism to avoid network congestion (C9). In [40], Liu, J., et al. have proposed an integrated networking scheme for MEC and fiber-wireless (FiWi) access networks. Their approach focuses on the dynamic orchestration of network, storage, and computing resources to meet diverse application demands, addressing the importance of mobility (C1) management. The lack of mobility data has been solved by collecting user's mobility information, such as locations and time via Access Points, Base Stations (BSs), or even sensors. Furthermore, the connectivity provided by FiWi access networks facilitates the direct communication of edge clouds without the core network, improving reliability (C7). The dynamically controlled routing enables efficient VM migration and service transmission link failure to ensure high reliability and availability of the services. Evaluations on an integrated scheme of edge clouds and multiple heterogeneous (C8) networks (e.g. FiWi) have been performed. Throughput (C9) levels have been verified between the edge and remote cloud servers. Video streaming services in MEC architectures have also been investigated in [41]. Yang, S., et al. have implemented a Proof-of-Concept (PoC) for a video streaming use case in a MEC-based architecture. It focuses on assessing MEPs to deploy novel applications at the edge, such as intelligent video accelerating services that need low latency and high bandwidth. The authors have developed two ML models for video and radio channel quality prediction to improve the overall QoE of video streaming users. These mechanisms improve service reliability (C7) and meet expected QoS levels. Their approach improves the radio channel quality (and thus the throughput) between mobile devices and the BS (C9). By deploying the video streaming service at the edge, their approach guarantees high throughput for most users. In [42], Shah, S., et al. have proposed the integration of SDN and cloud-native virtualization techniques to facilitate the orchestration and management of MEHs. Their work focuses on E2E mobility support to maintain service continuity when users relocate from one MEH to another (C1). Request/reply messaging patterns have been implemented based on the ZeroMQ protocol for inter-process communication between their SDN application and the target MEP. Protocols such as ZeroMQ and MQ Telemetry Transport (MQTT) have been designed to minimize network bandwidth and ensure reliability (C7). A heterogeneous (C8) Radio Access Technology (RAT) deployment has been considered in the evaluation of their Vehicle-to-Everything (V2X) use case. Multiple mobile operators and different SDN controllers have also been assessed (C10). The authors have also proposed an inter-slice resource sharing and federation model as future work. The authors aim to extend their work to support NS in their framework for mobility management between heterogeneous network slices across edge clouds.

Security in MEC is another open challenge that has been studied in [43]. Ranaweera, P., et al. propose MEC to enable security-as-a-service features. Experiments have determined the security functions scalability deployed in an MEP (C2). Security (C5) has also been discussed in detail. Security services are executed in Docker containers offering application-level isolation. The service deployment follows the SFC concept to dynamically adapt resources and the auto-scaling of security functions to accommodate several traffic profiles. The authors have also considered the inherent heterogeneity (C8) of IoT devices, claiming that the deployment of security services as a third-party solution is needed due to the strict requirements introduced by IoT. Their approach helps to mitigate security concerns and support heterogeneous mobile services through its flexibility. Lastly, Ksentini, A., et al. [44] have proposed the integration of both MEC and NS in a novel scheme compliant with ETSI and 3GPP specifications (the standardization bodies working on MEC and NS, respectively). A novel orchestration architecture has been presented, incorporating the MEC paradigm as a 5G sub-slice. Two different models to support NS in MEC have been proposed. On the one hand, a multi-tenancy model assumes that the MEP is deployed at the edge NFVI and is shared among the multiple slices. On the other hand, an in-slice deployment model considers that the MEP is deployed inside the slice (C4). In both models, the MEP is deployed as a Virtual Network Function (VNF). Security and privacy (C5) concerns have also been addressed. The authors state that each slice should not access the traffic or other information owned by other slices. For example, slices should not access information about the users of another slice, such as their location or channel quality. Multi-tenancy has been addressed, which is related to federation concepts, but from a single provider perspective instead of a multi-provider one. Cited works are adequate alternatives for security enhancement in future MEC infrastructures.

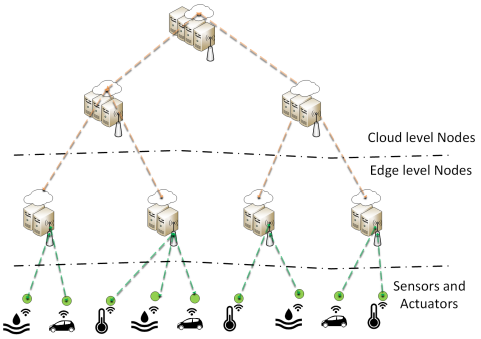


Fig. 4: High-level view of a Fog Computing environment [50].

### C. Fog Computing (FC)

Cisco has introduced the FC paradigm [61] in 2012 as an extension of cloud computing to provide resources on network edges to handle the massive growth of connected devices. Fig. 4 presents a high-level view of a hierarchical FC architecture. In contrast to a centralized cloud, fog nodes are distributed across the network to act as an intermediate layer between end devices and the cloud. These so-called fog nodes or edge locations are essentially small cloud entities that bring processing power, storage, and memory capacity closer to devices and end-users. This enables local operations, crucial for most IoT use cases to reduce the amount of data that needs to traverse the entire network up to the cloud. By deploying

services at the network edge, FC can also provide lower latency than traditional clouds. FC and MEC are close concepts [62] differing in the considered interactions (i.e. between edges and cloud): MEC deploys services close to end-users to reduce latency and avoid congestion in the network core, while FC considers bi-directional communications between edges and cloud due to the hierarchical architecture.

Novel FC architectures have been recently proposed, as in [45]. Sookhak, M., et al. have introduced a novel concept named Fog Vehicular Computing (FVC) to expand the computation and storage capacity of FC architectures. The authors describe a complete cross-layer architecture for FVC and introduce several components alongside a decision-making process for task scheduling. A Shopping Mall FVC use case has been evaluated, demonstrating the effectiveness of the proposed architecture. The feasibility of extending FC towards vehicles has been studied (C1). The evaluation has shown that deploying resources on vehicles could enhance a standard FC architecture. The authors state that their FVC architecture aims to improve the scalability of an FC infrastructure (C2). Different hardware capacities have been considered for fog nodes, which satisfies heterogeneity (C8). Throughput (C9) has also been evaluated in the context of communication costs at the fog layer. Their work compares an FC and their FVC architectures. The authors intend to focus on user security and privacy as future work, especially in terms of secured data access control and data encryption in FVC. In [46], Moreno-V., R., et al. have presented a Hybrid Fog and Cloud (HFC) framework to optimize the automated provisioning of virtual networks to connect geographically distributed fog and cloud sites. The approach adopts an agent-based solution allowing interaction with different cloud providers and fog infrastructures while providing scalability, security and multi-tenancy. The authors state that L2 overlay networks present advantages over L3 overlay networks, including native support for mobility (C1) and migration. Hosts in different sites will share a common overlay addressing scheme, enabling host migrations with minimal reconfiguration. Their HFC framework supports different topologies (e.g. tree or mesh), enabling the addition or removal of a fog or cloud location with minimal configuration (C2). The authors have also stated that it is necessary to guarantee data privacy and integrity in a hybrid environment (where the interconnection of different cloud and fog sites runs over public networks) by implementing the overlay networks over secure communication channels (C5). The authors remark that the reliability (C7) of the overlay network can be increased if HFC agents are deployed in a high-availability cluster that shares a common routing public IP. The authors support their HFC framework stating that tenants can deploy a virtual network over heterogeneous fog infrastructures and clouds (C8). A PoC of the HFC framework has been implemented to assess the throughput (C9) of L2 and L3 overlay networks. The HFC framework has also been designed to support multiple application providers and tenants (C10). Each provider can instantiate its applications and virtual networks to provide services to its end-users. Both works propose suitable FC-based architectures to manage a large number of connected devices and support low latency services.

Several works have also studied the integration of SDN concepts into FC to overcome scalability issues. In [47], Sharma, P. K., et al. have presented a distributed cloud architecture based on SDN and BC for secure and on-demand access to IoT services. The authors adopt SDN and BC to design a highly scalable architecture for IoT (C2). Leveraging BC technology, user privacy can be assured since third-party entities do not access or control user data (C5). Each user manages its own security keys, and each node stores only encrypted fragments of user data. Simulations have evaluated the accuracy of their architecture in detecting and mitigating saturation attacks at the edge of the network. Moreover, the authors state that if nodes fail, the computation should continue on another node to maintain resilience (C6) levels. The authors have also implemented a scheduling algorithm to match users' preferences, with reliability (C7) as a decisive factor. In the evaluation, nodes with different computing capacities have been considered (C8). Evaluations have assessed the delay, response time, throughput (C9), and the ability to detect real-time attacks of their approach. Results have shown that their approach provides higher throughput than traditional cloud infrastructures. The authors propose to focus on energy-efficient communications for edge devices as future work. In [48], Bruschi, R., et al. have discussed an SDN-based slicing scheme for multi-domain fog-cloud services. The approach has been designed with high scalability (C2) in mind, minimizing the number of OpenFlow (OF) rules in the overlay implementation. Their SDN scheme implements NS to provide service isolation (C4). Heterogeneity (C8) has often been addressed (e.g. nodes geographical distribution, different QoS requirements). Solving scalability concerns will definitively help adopting these novel architectural paradigms.

IoT and Smart City services have been thoroughly studied in the FC domain. In [49], Baccarelli, E., et al. have proposed to integrate FC and IoE, describing the main building blocks of a Fog of Everything (FoE) platform. The authors state that fog nodes should be arranged into spatial clusters to serve mobile devices through single-hop links to deal with high mobility patterns (C1). The authors state that the fog layer of the FoE platform should handle resource scalability (i.e. computing, storage, and network) of most big data applications (C2). The authors suggest that inter-device communications should occur through Device-to-Fog (D2F) links in place of Device-to-Device (D2D) links to reduce energy consumption (C3). The performance of the FoE architecture in terms of delay and energy, compared to a typical D2D architecture, has been evaluated. Results have shown that FoE achieves lower delays and higher energy efficiency. The authors also propose to implement Transmission Control Protocol (TCP) NewReno [63] to guarantee reliable E2E connections, even when facing network congestion or link failures (C7). Heterogeneous devices and network topologies have been assessed (C8). The energy assessment of the FoE architecture has also been made based on the average throughput (C9) of all established connections (e.g. F2D). Lastly, in [50], an FC framework for autonomous management and orchestration in 5G-enabled Smart Cities has been proposed. The approach follows the guidelines of ETSI NFV MANO, extending it with software components towards

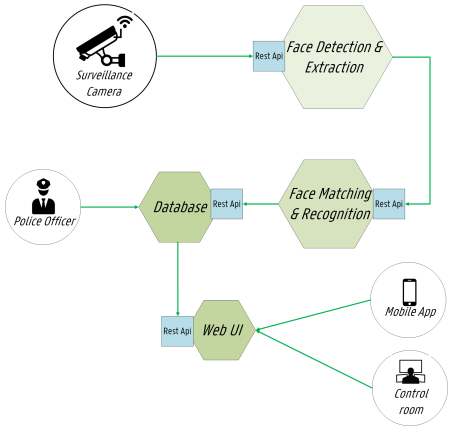



Fig. 5: An example of a Surveillance camera use case based on the micro-service architecture.

a fully integrated fog node management system. A Peer-to-Peer (P2P) fog protocol has been presented to exchange application service provisioning information between fog nodes. The work also follows guidelines defined by ETSI oneM2M [64] to monitor mobility patterns (C1) and provide proper device management and security (C5) functionalities. FC enables distributed malware monitoring tools to compensate for IoT devices' limited security and detect threats and attacks on time. Energy efficiency (C3) has also been discussed in the context of resource-constrained IoT devices and proper resource allocation. A distributed anomaly detection approach for FC has also been introduced to avoid transmissions of incorrect information and improve network reliability (C7). This anomaly detection scheme has been evaluated based on a heterogeneous (C8) fog-cloud environment. The cited works address important challenges for low latency service delivery by focusing on IoT concerns: mobility support, network reliability and interoperability between heterogeneous devices and environments.

### *D. Micro-services*

Recently, micro-service patterns [65] have gained tremendous attention. Container-based services revolutionized the way developers build their applications. An application is decomposed into a set of loosely coupled services that are developed, deployed and maintained independently. Each service is responsible for a single task and communicates with the other services through lightweight protocols. These services can be developed in different programming languages and even using different technologies. Containers are currently the most promising alternative to the traditional monolithic application paradigm, mostly centralized and code-heavy. Containers are the main alternative to conventional VMs due to their low overhead and high portability. Fig. 5 presents a high-level view of a Surveillance camera use case envisioned for Smart Cities based on the micro-service paradigm.

Security has been a major concern in micro-service architectures. In [51], Brenner, S., et al. have focused on integrating trusted execution based on Intel Software Guard Extensions (SGX) into micro-services. Their approach increases privacy and data confidentiality to sensitive applications deployed through micro-services (C5). By integrating SGX into the micro-service toolkit, the authors achieve higher levels of security. Results have proved the feasibility of their approach while showing low-performance overheads. Response time and throughput (C9) evaluations demonstrate equivalent throughput levels between secured micro-services and regular ones. In [52], Guija, D., et al. have proposed a 5G platform with integrated authentication and authorization features for micro-services in an NFV environment. Their architecture adopts the NFV-based SONATA Service Platform, which offers continuous integration and management of the entire VNFs life cycle. Keycloak [66], an Identity and Access Management open-source tool, has been proposed to isolate each sub-component in their architecture and ensure higher scalability (C2). The authors note the importance of sub-component isolation (C4) while proposing several authentication and authorization mechanisms (C5). A user management module has been presented to handle identities, permissions, and authorizations, allowing or denying operations to users or internal components. Cited works show that micro-services can efficiently secure and isolate future applications.

Architectural enhancements have been addressed in [53]. Xu, R., et al. have presented a BC-based decentralized architecture called BlendMAS for IoT-based public safety systems. Authentication and access control models have been transcoded into smart contracts deployed on private BC networks. All security mechanisms aim to improve system security and offer higher scalability (C2) and flexibility. The authors state that leveraging BC technologies establishes secure user relationships over computer networks (C5). A PoC of the BlendMAS architecture has been evaluated based on a surveillance use case where micro-services are deployed on distributed edge and fog nodes. A distributed network environment with a large number of heterogeneous (C8) devices (i.e. cameras, laptops, desktops, Raspberry Pis) has been considered. In [54], Debauche, O., et al. have presented an edge-based architecture to deploy AI algorithms and models for IoT. The architecture has been implemented in Kubernetes [67], a well-known container orchestration platform. Their architecture provides several benefits such as lower latency and higher scalability (C2) than traditional cloud infrastructures. It also supports several different types of micro-services, enabling ML features at the edge (C8). The authors propose studying cluster federation as future work. Micro-services promise to relieve the burden of costly service deployments from traditional VMs.

### *E. Hardware Acceleration*

Hardware acceleration [68] has become a promising research field to mitigate performance degradation and latency introduced by network softwarization. Softwarized NFs have revolutionized network infrastructures by providing higher flexibility, higher portability, and reconfigurability. However, migrating conventional hardware functions towards softwarized VNFs is challenging. Recent works address current barriers to meet the flexibility and scalability demands of

modern communication networks. In [55], Zhang, X., et al. have designed a hardware-based approach for NFV. It provides high scalability (C2) and programmability while supporting hardware-level parallelism and reconfiguration. The authors state that it is important to install each software component into a VM to isolate (C4) customized NFs. Their platform consists of heterogeneous (C8) middleboxes adopting both FPGAs and microprocessors to implement NFV operations, dynamically customizing specific network flow needs. Latency and throughput (C9) of both FPGA and VM module implementations have been evaluated.

Recently, hardware-accelerated platforms have focused on Neural Networks (NNs). In [56], Umuroglu, Y., et al. have presented Finn, a framework for building fast and flexible FPGA accelerators. The authors adopt binarized NNs to achieve higher performance in Tera Operations per second (TOPS) on FPGAs. Their evaluation demonstrates the performance and energy efficiency (C3) of binarized NNs for image classification. A heterogeneous (C8) streaming architecture has been designed, in which a custom architecture is built for a given topology rather than scheduling operations on top of a fixed architecture. Separate compute engines are dedicated to each layer, communicating via on-chip data streams. Results have shown benefits in classification throughput (C9), FPGA resource usage and power consumption. In [57], Cai, R., et al. have proposed VIBNN, an FPGA-based accelerator for bayesian NNs. A deep pipelined architecture achieving high scalability (C2) and efficient memory access has been designed. Results have shown that VIBNN can reduce energy consumption (C3) while attaining high throughput (C9) levels. Lastly, Owaida, M., et al. [58] have explored an FPGA-based accelerator to improve the overall performance of data processing pipelines. The authors focus on decision tree ensemble methods, a common approach to score and classify search systems. Results have proved the high scalability (C2) and throughput (C9) of their approach. Hardware impacts the performance of softwarized NFs. All cited works show adequate enhancements to support future VNFs.

### F. Summary

Section III introduces novel research on architectural paradigms enabling the deployment of service chains on a continuum of virtual resources. ETSI MEC has been designing a reference architecture for future mobile networks, creating a cloud environment close to the RAN while FC is placing resources at the edge to meet the strict requirements of IoT. Micro-services are transforming service deployments from a traditional monolith to loosely-coupled containers, while hardware-based accelerators are pushing the boundaries of hardware platforms to support softwarized NFs. The literature review has shown differences in the evaluation criteria. MEC and FC mainly discuss *Mobility* (C1), while FC, micro-service and hardware acceleration focus on *Scalability* (C2). Mobility support and service migration are open challenges in MEC and FC. Without efficient migration strategies, low latency service delivery cannot be achieved. *Energy efficiency* (C3) and *Isolation* (C4) are unexplored in MEC and FC, while *Security* (C5) is the main focus of micro-service research. Micro-services establish security guarantees while offering flexible service deployments. *Resilience* (C6) is unexplored in all domains, while FC and MEC address *Reliability* (C7). All domains address *Heterogeneity* (C8), while *Throughput* (C9) is more noticeable in FC and hardware acceleration. Hardware-based platforms improve the performance of softwarized NFs, supporting high throughput levels while attaining low latency. *Federation* (C10) concepts are unexplored, though authors acknowledge their importance.

## IV. Recent advances on communication networks for low-latency services

### A. Overview

Novel networking paradigms have opened several possibilities for improving network performance, including higher flexibility and scalability. Recent trends bring software-based automation to current networks: NS, SFC, IBN and SR. Table V presents a summary of the reviewed works.

### B. Network Slicing (NS)

NS [91] implements independent E2E logical networks on top of physical infrastructures. A slice is a virtual network implemented on top of physical nodes, creating the illusion of operating its dedicated physical network. NS allows high flexibility, improved resource allocation and increased service isolation by physically separating network resources. NS has gained significant attention with the advent of 5G. In [69], Campolo, C., et al. have designed 5G network slices for V2X services, addressing mobility (C1) management for V2X slices. The authors state that mobility prediction models help to optimize caching strategies for vehicles. Isolation (C4) strategies have been discussed, such as intra-slice and inter-slice V2X isolation. The authors remark that life cycle management configuration, adaptation, and monitoring are essential to meet slice isolation constraints and QoS levels. Retransmissions are handled locally by each vehicle in autonomous driving slices to match high-reliability and ultra-low-latency requirements (C7). Heterogeneity (C8) has been satisfied through different requirements (e.g. latency, bandwidth) for different slice types (e.g. URLLC, xMBB). Throughput (C9) has been discussed for distinct slices. The authors state that large Transmission Time Intervals (TTIs) (e.g. 1 ms) should be used for throughput-demanding applications, while short TTIs (e.g. 0.125 ms) can be used for fast retransmissions in teleoperated driving slices. In [70], Ksentini, A., et al. have proposed a framework to enforce NS in RAN. It focuses on separating network traffic towards the appropriate core and uses a two-level scheduler to adapt the resource allocation policy according to the slice's needs. Their architecture shares the usage of logical channels and their mapping to Evolved Packet System (EPS) bearers with legacy LTE. The main difference lies in abstracting physical resource blocks through a slice resource manager responsible for allocating resources for each User Equipment (UE) belonging to its slice (C4). The authors state two important factors for resource allocation in URLLC slices: latency and reliability (C7). To maximize the

TABLE V: Summary of the reviewed works in terms of Network.

| Research Domain | Authors | Main focus | Year | Evaluation Criteria | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | C10 |
| Network Slicing (NS) (Sec. IV-B) | Campolo, C., et al. [69] | Vehicle Networks | 2017 | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ |
| | Ksentini, A., et al. [70] | Slicing for RAN | 2017 | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ |
| | Taleb, T., et al. [71] | 5G use cases | 2017 | ✓ | ✓ | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ |
| | Popovski, P., et al. [72] | 5G use cases | 2018 | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ |
| Service Function Chaining (SFC) (Sec. IV-C) | Qu, L., et al. [73] | Reliability | 2017 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ |
| | Zhang, L., et al. [74] | Network Coding | 2017 | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ |
| | Trajkovska, I., et al. [75] | SDN-based SFC | 2017 | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| | Jang, I., et al. [76] | SFC placement | 2017 | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Bhamare, D., et al. [77] | SFC placement | 2017 | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ |
| | Hawilo, H., et al. [78] | SFC placement | 2019 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ |
| | Xiang, Z., et al. [79] | Tactile Internet | 2019 | ✗ | ✓ | ✗ | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | ✗ |
| Intent-based Networking (IBN) (Sec. IV-D) | Cerroni, W., et al. [80] | IBN orchestration | 2017 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✓ |
| | Arezoumand, S., et al. [81] | SDN-based IBN | 2017 | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ |
| | Szyrkowiec, T., et al. [82] | SDN-based IBN | 2018 | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Abbas, K., et al. [83] | Slicing for IBN | 2020 | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| | Wang, Y., et al. [84] | Privacy | 2020 | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ |
| Segment Routing (SR) (Sec. IV-E) | Pang, J., et al. [85] | SDN-based SR | 2017 | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Giorgetti, A., et al. [86] | Multi-domain SR | 2017 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ |
| | Cianfrani, A., et al. [87] | SR performance | 2017 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Desmouceaux, Y., et al. [88] | Load balancing in SR | 2018 | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✓ | ✗ |
| | Chunduri, U., et al. [89] | SR scalability | 2018 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| | Aubry, F., et al. [90] | SR performance | 2018 | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ |

latter, the authors suggest adapting the modulation and coding scheme used by UEs to improve robustness to channel errors. Thus, robust modulations should be favored over high data rate modulations. Heterogeneity (C8) has also been studied through different slice requirements (e.g. latency, bandwidth, reliability). Throughput (C9) has also been evaluated. Results show that Extreme Mobile Broadband (xMBB) slices achieve the highest throughput while URLLC slices achieve the least levels since this slice focuses on maximizing reliability. The authors propose to study the scalability of the two-level scheduler as future work.

5G use cases have been customized based on NS. In [71], Taleb, T., et al. have personalized mobile networks at different granularity levels (e.g. application, network, group of users). An architecture named PERMIT has been presented, considering user mobility (C1), usage behavioral patterns, and underlying dynamics of the infrastructure for service customization. The authors state that users are aggregated in the same slice when sharing a service or behavior due to scalability (C2) reasons, and to isolate (C4) usage profiles and avoid security (C5) breaches. In [72], Popovski, P., et al. have studied non-orthogonal sharing of RAN resources in uplink communications from a set of enhanced Mobile Broadband (eMBB), massive Machine-Type Communication (mMTC), and URLLC devices to a common BS. NS has been investigated for RAN access (C4). A communication-theoretic model has been presented, considering heterogeneous requirements for the three services (C8). The reliability diversity concept has also been introduced as a design principle that leverages reliability requirements across all services, ensuring the performance of non-orthogonal RAN slicing (C7). Reliability and throughput (C9) levels have been evaluated. 5G has encouraged academia and industry to implement NS for future use cases, allowing service personalization and isolation.

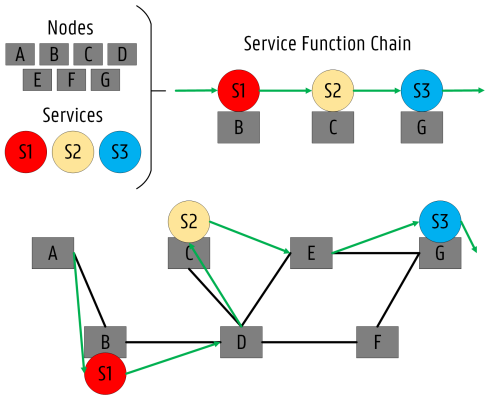



Fig. 6: An example of a service chain deployment [92].

## C. Service Function Chaining (SFC)

SFC [93], [94] has been studied in network management over the last few years. A service chain concerns proper service ordering. Fig. 6 shows how each user has to traverse the service chain to access a network service. The circles represent different services while the arrows show how traffic is steered in the network. User requests are routed through the service chain following a service graph, which aims to improve resource allocation and application performance. SFC is a flexible and reliable alternative to dynamically reconfigure softwarized services without replacing hardware. SFC has been recently applied to reduce latency in softwarized VNFs. In [73], Qu, L., et al. have proposed a reliability-aware provisioning approach with delay guarantees for NFV-enabled Data Center (DC) networks. A Mixed-Integer Linear Programming (MILP) formulation optimizes VNF placement and traffic routing focused on maximizing reliability and reducing E2E delays. A heuristic has also been introduced to overcome the MILP complexity and consequent high execution time (C2). Several

constraints satisfy reliability (C7) guarantees considered by the authors. Throughput (C9) has been evaluated for both methods. Results have shown that their heuristic outperforms existing schemes in average E2E delays and reliability at the expense of additional bandwidth and resource usage. In [74], Zhang, L., et al. have designed and implemented network coding as an NF in VMs for geo-distributed cloud DCs, proposing efficient algorithms for deploying and scaling network coding functions. The authors aim to improve network reliability (C7). Results have shown increased throughput (C9) and higher robustness of multicast sessions. Also, SFC concepts have been adopted in SDN-based approaches as in [75]. Trajkovska, I., et al. have proposed an SDN-based SFC mechanism with performance and scalability (C2) in mind, designed to ensure tenant isolation (C4). A prototype has been evaluated in a real DC, where the impact of heterogeneous (C8) environments has been assessed. Throughput (C9) levels have been evaluated for distinct chain scenarios based on a video traffic use case. Low latency SFC is essential for future applications. Cited works have developed adequate mechanisms to ensure SFC performance and scalability.

Another challenge is SFC allocation and placement. In [76], Jang, I., et al. have studied optimal SFC allocation and flow routing. A multi-objective MILP model has been proposed to maximize the acceptable flow rate and to minimize energy costs for multiple service chains. A polynomial-time algorithm based on linear relaxation has also been presented to approximate the optimal solution provided by the MILP model. The energy cost minimization has been modeled as one of the objectives (C3). Results have shown that their polynomial algorithm obtains near-optimal performance and increases the acceptable flow rate (i.e. throughput (C9)) and service capacity compared to other algorithms. In [77], Bhamare, D., et al. have presented an ILP model for SFC allocation in a multi-cloud scenario. An Affinity-based approach (ABA) has been proposed for large networks. Deployment costs have been considered in the formulation, leading to energy savings (C3). Their heuristic has been compared to greedy algorithms. Results have shown that the ABA algorithm outperforms greedy heuristics in total delays and total resource cost. The evaluation assesses traffic loads (C9) and considers multi-cloud environments (C10). In [78], Hawilo, H., et al. have proposed a MILP model and a heuristic algorithm for VNF placement. The authors study the carrier-grade nature of NFV applications and the minimization of E2E delays in service chains. Their evaluation considers scalability (C2) requirements. The authors state that if delay constraints are violated, the scalability and the traffic offloading capacity of the service chain are affected. The approach enhances the reliability (C7) and the QoS of the service chain by maximizing the number of participating members in a functional group of a VNF instance. Heterogeneity (C8) has also been satisfied through heterogeneous VNF structures and distinct placement requirements (e.g. scalability, E2E delay). SFC has also been studied for latency reduction in future use cases, such as Tactile Internet in [79]. Xiang, Z., et al. have investigated the feasibility of NFV concepts to offer low latency for Tactile Internet. The authors have designed a management framework for distributed SFC in MEC, implementing multiple VNFs in parallel to evaluate its scalability (C2) and flexibility. Their framework considers a virtualized networking overlay on top of the physical infrastructure, where multiple VMs are connected to and isolated from each other (C4). The authors adopt encryption methods to ensure data confidentiality and integrity (C5). Heterogeneity (C8) has been addressed through different networking components (e.g. physical networks, virtual overlays). The trade-off between per-packet latency and throughput (C9) has been evaluated. Their approach reduces packet throughput to achieve shorter per-packet latency, required for typical Tactile Internet applications with a 1 ms round-trip delay budget. Efficient SFC placement and routing is currently the main challenge. Solving these issues will lead to flexible service deployments supporting low latency services throughout their execution.

### *D. Intent-based Networking (IBN)*

IBN [95] has been recently proposed to communicate intents to the network. Intents are policies written in high-level operational or business objectives that a system should meet. The main idea behind IBN is to communicate to the system how it should behave without detailing how it could achieve the objective. The intent is enforced in the infrastructure by the system, free from human intervention. An Internet Engineering Task Force (IETF) working group has been defining concepts, specifications and functionalities of IBN [96]. IBN aims towards autonomous networks, simple to manage with minimal human intervention powered by ML and AI. IBN concepts have been recently incorporated into management and orchestration practices, including SDN-based platforms. In [80], Cerroni, W., et al. have proposed a reference architecture and an intent-based Northbound Interface (NBI) for E2E service orchestration across multiple domains. The scalability (C2) of the NBI response time has been assessed at the Virtual Infrastructure Manager (VIM) implemented in ONOS [97], an open-source SDN controller. Two QoS features have been evaluated: low latency and high reliability (C7). The approach has been validated in a heterogeneous (C8) multi-domain (C10) testbed (i.e. IoT, OpenFlow, and cloud) based on an IoT use case. In [81], Arezoumand, S., et al. have presented an intent framework named MD-IDN for multi-domain cloud infrastructures. Compilation algorithms have been proposed to achieve high scalability (C2) in multi-domain networks. The authors remark that isolation (C4) is crucial in multi-domain environments, thus, intent frameworks must avoid cross-contamination of intents requested by different tenants. Their algorithms have been assessed over heterogeneous (C8) and multi-domain (C10) networks. Distinct infrastructural nodes have been considered (e.g. VMs, FPGAs, and GPU servers). Results have shown that the MD-IDN framework outperforms current practices that compile intents over a flat network topology. In [82], Szyrkowiec, T., et al. have presented an architecture for automatic intent-based provisioning of security services through a multi-layer SDN Orchestrator. Their orchestrator defines a lightweight NBI, specifying application needs focused on security (C5)

configurations. Encryption layer properties have been analyzed regarding latency, throughput (C9), flexibility, and protocol transparency. Cited works show that IBN revolutionizes orchestration practices, enabling highly automated networks.

IBN concepts have also been combined with NS in [83]. Abbas, K., et al. have designed an IBN slicing system to manage core and RAN resources. Users can set intents and their system configures the network accordingly. A DL model for resource management has also been presented. NS has been applied together with IBN to efficiently handle RAN and core resources in 5G networks (C4), considering different slices (i.e. eMBB, IoT, and URLLC) (C8). The uplink and downlink throughput (C9) achieved by the three slices have been assessed. Lastly, privacy concerns have been addressed in [84]. Wang, Y., et al. have presented an intent prediction-based approach named LocJury to preserve location privacy in Internet-of-Vehicles (IoV). Their method estimates the intent of location access and restricts malicious attempts. Mobility (C1) plays a major role in preserving location information in IoV. LocJury applies ML and IBN to learn the motivation behind location accesses and restrain suspected malicious attempts. Results have shown that LocJury preserves vehicles' location privacy (C5).

### *E. Segment Routing (SR)*

SR [98] leverages the source routing paradigm. A node steers a packet through an ordered list of instructions, called segments. A segment can be composed of any type of instruction (e.g. topological or service-based information), which is then placed as path state information into a packet header at an ingress node. Several segments create unconstrained network paths represented by segment lists. Information flows from packet headers, making nodes stateless and significantly reducing forwarding tables complexity, simplifying traffic engineering and management across network domains. SR is highly responsive to network changes, making networks more agile and flexible. SR has already been explored in SDN and multi-domain scenarios. In [85], Pang, J., et al. have presented a collaboration method of multipath TCP (MPTCP) and SR to address resource consumption in an SDN-based DC Network (DCN). The authors combine MPTCP and SR to reduce forwarding rules required in SDN-based MPTCP solutions. The authors prove that their approach cannot meet the increasing transmission demand in highly scalable scenarios (C2) in a single-controller mode due to the packet header size limitation in SR. In contrast, it can significantly reduce storage, minimizing energy consumption (C3). Simulations have shown that high throughput (C9) is achieved while reducing flow completion time and improving link resource usage in SDN-based DCNs. The authors propose to investigate multi-controller environments as future work. In [86], Giorgetti, A., et al. have focused on two relevant SR use cases: dynamic traffic recovery and traffic engineering in multi-domain networks. The scalability (C2) of the segment list depth has been assessed. The experiments consider a multi-domain (C10) heterogeneous (C8) testbed exploring an SDN-based implementation of SR.

Performance and scalability are major concerns in SR. In [87], Cianfrani, A., et al. have faced the challenge of transitioning from a pure IP network to a full SR system while optimizing network performance. The authors have proposed an architecture called SR Domain (SRD) to ease the coexistence between IP routers and SR nodes. A MILP formulation has been introduced for the SRD design problem by reducing congestion. Their performance study has shown that the SRD approach can reduce the maximum link usage and increase system stability. Consequently, higher scalability (C2) is achieved. Heterogeneity (C8) has also been studied in hardware routers (IP routers and SR nodes). In [88], Desmouceaux, Y., et al. have presented the concept of 6LB, a load balancer running exclusively within the IP forwarding plane. It applies IPv6 SR to direct data packets from a new flow through a chain of candidate servers. The authors propose a consistent hashing algorithm and an in-band stickiness protocol to reliably distribute 6LB across several instances for scalability purposes (C2). Simulations have assessed the consistent hashing algorithm resiliency (C6). By adopting the power of two choices [99], 6LB significantly increases reliability (C7) compared to single-choice approaches. Packet-forwarding performance assessment shows that higher fairness comes at a negligible cost of CPU overhead. Throughput (C9) levels have been compared to single choice load-balancing approaches. In [89], Chunduri, U., et al. have designed the Preferred Path Routing (PPR) concept to overcome SR limitations. PPR minimizes the data plane overhead (e.g. packet processing) by extending it for IP data planes without replacing existing hardware or even upgrading the data plane. PPR allows dynamic path QoS reservations by providing deterministic queuing latency. Scalability concerns (C2) have been discussed since PPR requires a separate PPR-ID for every possible path. The authors state that they do not expect deployment issues in a practical setting since the total number of preferred paths can be easily supported by network devices. The authors also propose studying fast rerouting and path resiliency schemes as future work. In [90], Aubry, F., et al. have introduced Robustly Disjoint Paths (RDPs): pairs of paths that remain disjoint even after an input set of failures, with no external intervention. The authors have designed efficient algorithms to compute SR-based RDPs. Evaluations on real network topologies have shown that RDPs achieve high scalability (C2) and reliability (C7) in large ISP networks. Fault tolerance (C6) has been assessed through single and multiple link failure experiments. Novel routing paradigms aim to improve the reliability and resilience of current networks. Cited works prove that SR is suitable for addressing multi-domain challenges in future networks.

### *F. Summary*

Section IV presents novel research paradigms on communication networks. NS enables higher levels of flexibility and isolation, while SFC optimizes resource allocation through proper service ordering. IBN communicates intents to the network, thus, enforcing rules without detailing how the system should obtain them. SR revolutionizes the routing paradigm since it

leverages the source routing paradigm to move information in the network in packet headers, making nodes stateless. The literature review shows that NS discusses *Mobility* (C1), while SFC focuses on *Scalability* (C2) and *Energy efficiency* (C3). NS also deals with *Isolation* (C4). Both NS and SFC are important for low latency service delivery. On the one hand, NS allows the setup of different QoS levels for distinct applications. On the other hand, SFC brings high degrees of flexibility and reconfiguration, optimizing service placement and traffic flow. IBN addresses *Security* (C5) concerns, while SR studies *Resilience* (C6). SR and IBN enable network automation features that will ensure proper management and orchestration of multi-domain environments. Both concepts enable high automation but also attain high performance and scalability, fully supporting low latency services. *Reliability* (C7) has been explored in NS, SFC and SR, while *Heterogeneity* (C8) has been discussed in NS, SFC and IBN. SFC also focuses on *Throughput* (C9). *Federation* (C10) is still unexplored in these research domains, but authors acknowledge its importance in multi-domain environments.

## V. Towards efficient orchestration in cloud-native infrastructures

### A. Overview

Cloud-native infrastructures have imposed strict specifications on management functionalities. This section presents trends on orchestration: resource allocation, auto-scaling, performance monitoring and caching. Table VI summarizes the reviewed works.

### B. Resource Allocation

Resource allocation, also known as resource provisioning, has been studied for years in the network management domain [123], [124]. It concerns the provisioning of computing, network and storage resources required to instantiate services requested by users and devices over the Internet. Recently, cloud providers and users have been working together towards an efficient allocation of computing resources. Users expect the best QoS at the cheapest rate while cloud providers aim to increase their revenue and respect the agreed QoS level. With the advent of IoT and low latency services, resource allocation has become even more important. Delay-sensitive services (e.g. connected vehicles, XR) require latency in the order of milliseconds that centralized infrastructures cannot support, thus, requiring efficient allocation in a continuum of virtual resources. Allocation strategies for vehicular networks and IoT contexts have been studied. In [100], Liang, L., et al. have investigated spectrum sharing and power allocation for D2D vehicular networks. The authors state that fast channel variations caused by high mobility (C1) in a vehicular environment need to be considered in allocation scheme design. Also, different QoS requirements for Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) links have been studied. The authors state that high link capacity is desired for V2I connections, while safety-critical information of V2V connections places greater emphasis on link reliability. Their scheme includes reliability (C7) guarantees for D2D users. The heterogeneous (C8) performance of V2I and V2V links for resource allocation purposes has been studied. The maximization of the overall V2I link throughput (C9) has been included as an optimization objective. In [101], Arkian, H. R., et al. have presented a fog-based scheme called MIST for cost-efficient resource allocation of crowdsensing applications in IoT. Firstly, the authors propose a Mixed-Integer Nonlinear Programming (MINLP) model. To tackle its inherent high computational complexity, the MINLP model has then been linearized into a MILP formulation. The authors have studied the joint optimization of data consumer association, task distribution, and VM placement issues towards minimizing the overall cost (C3) while satisfying QoS levels. Heterogeneity (C8) has been satisfied through distinct fog nodes (e.g. routers, access points) and different wireless connectivity (e.g. 4G and Wi-Fi). In [102], Santos, J, et al. have presented an ILP formulation for IoT service placement. The model considers multiple optimization objectives, such as low latency and energy efficiency (C3). Smart City use cases with different placement requirements and a fog-cloud infrastructure with distinct hardware capabilities have been considered (C8). In [103], Yao, J., et al. have addressed the joint optimization of resource allocation and power control in FC to minimize the overall system cost while satisfying QoS requirements. The authors have formulated the problem as an MINLP model and then presented an approximation algorithm to solve it. The authors discuss mobility (C1) management in IoT when maintaining QoS levels. To manage device mobility in fog-aided networks, the authors state that its transmission power can be adapted and that handovers between different IoT gateways and VM migrations should be performed to meet QoS requirements. Simulations have evaluated power control and system cost (C3), based on multiple applications with different QoS requirements (C8).

Resource allocation has also been investigated in SDN and access control in mobile networks. Podili, P., et al. [104] have introduced a resource provisioning approach for virtual networks in SDN focused on E2E delay and bandwidth. Results have shown improvements in availability, scalability (C2), and cost-effectiveness (C3). The authors propose a destination label forwarding mechanism to reduce the number of flow rules in SDN switches: a unique set of labels is assigned to each virtual network, thus, ensuring traffic isolation (C4). In [105], Zhang, H., et al. have studied the problem of energy-efficient user scheduling and power optimization in Non-Orthogonal Multiple Access (NOMA) networks. The trade-off between data rate performance and energy consumption (C3) has been assessed for wireless downlink communications in heterogeneous (C8) NOMA networks. Results have demonstrated improved energy consumption and user throughput (C9). In [106], Zhou, Z., et al. have proposed a two-stage access control and resource allocation algorithm for Machine-to-machine (M2M) communications in industrial automation. Firstly, a contract-based mechanism motivates delay-tolerant devices to postpone their access voluntarily. Then, a long-term cross-layer online resource allocation model jointly optimizes rate control, power allocation, and channel selection without prior knowledge of channel states. Results have shown im-

TABLE VI: Summary of the reviewed works in terms of Orchestration.

| Research Domain | Authors | Main focus | Year | Evaluation Criteria | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | C10 |
| Resource Allocation (Sec. V-B) | Liang, L., et al. [100] | D2D communications | 2017 | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ |
| | Arkian, H. R., et al. [101] | IoT applications | 2017 | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Santos, J, et al. [102] | IoT applications | 2017 | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Yao, J., et al. [103] | IoT applications | 2018 | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Podili, P., et al. [104] | QoS in SDN | 2018 | ✗ | ✓ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| | Zhang, H., et al. [105] | Energy efficiency | 2018 | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| | Zhou, Z., et al. [106] | M2M communications | 2019 | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ |
| | Farhad, A., et al. [107] | Mobility Management | 2020 | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ |
| Auto-scaling (Sec. V-C) | Aslanpour, M. S., et al. [108] | Web applications | 2017 | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Rahman, S., et al. [109] | VNFs | 2018 | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| | Lee, D., et al. [110] | SFC | 2020 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Lin, T., et al. [111] | QoS | 2020 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✓ |
| Performance Monitoring (Sec. V-D) | Moradi, F., et al. [112] | Containers | 2017 | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Tangari, G., et al. [113] | Decentralized SDN | 2017 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| | Shah, S. Y., et al. [114] | Cloud applications | 2017 | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ |
| | Perdices, D., et al. [115] | Network Performance | 2018 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Sanz, I. J., et al. [116] | SFC performance | 2018 | ✗ | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ |
| Caching (Sec. V-E) | Chen, M., et al. [117] | Mobility-aware in 5G | 2017 | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ |
| | Tantayakul, K., et al. [118] | Mobility Management | 2017 | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ |
| | Zhang, K., et al. [119] | 5G MEC | 2018 | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Hao, Y., et al. [120] | Energy efficiency for MEC | 2018 | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Xiao, L., et al. [121] | Security | 2018 | ✗ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ |
| | Cheng, F., et al. [122] | Security | 2019 | ✓ | ✗ | ✓ | ✗ | ✓ | ✗ | ✗ | ✗ | ✓ | ✗ |

provements in sensing rate, queue stability, backlog fluctuation and energy efficiency (C3). The reliability of M2M communications has also been studied in device battery life as a long-term average power consumption constraint in their model (C7). Their scheme has been compared with the snapshot-based throughput optimal algorithm (as baseline), which maximizes physical-layer throughput (C9) without considering long-term constraints and sensing rate control. Lastly, mobility management has been studied in [107]. Farhad, A., et al. have presented a mobility-aware allocation scheme for IoT devices (C1) that enhances the Packet Success Ratio (PSR) by reducing the impact of interference, retransmissions, and packet loss compared with the LoRaWAN-based Adaptive Data Rate (ADR). Energy consumption (C3), PSR and reliability (C7) have been evaluated. Their scheme considers traffic heterogeneity (i.e. static and mobile end devices) based on the gateway sensitivity during the initial deployment phase (C8). Simulations demonstrate the feasibility of their scheme for IoT mobile applications needing high PSR and reliability without high energy consumption. Cited works tackle resource allocation starting from different points of view: several use cases and multiple requirements.

### C. Auto-scaling

Distributed clouds have revolutionized resource management. On the one hand, if services need more resources (i.e. under-provisioning), then they should be added on-demand so that services keep operating. On the other hand, resources should be released when they are not fully used (i.e. over-provisioning). Service over-provisioning wastes resources and increases costs, while under-provisioning schemes degrade performance and violate Service Level Agreements (SLAs). Thus, automatic mechanisms should scale up and down resources according to the network demand without human intervention. This is called Auto-scaling [125], where resources are dynamically added or removed to meet QoS requirements. Designing efficient auto-scaling systems is not a trivial task due to limited hardware resources, dynamic workloads, diverse service requirements and complex infrastructures.

Auto-scaling for web applications has been studied in [108]. Aslanpour, M. S., et al. have proposed a cost-aware auto-scaling mechanism (C2). Their approach focuses on executing scale-down commands via the selection of surplus VMs, which are then quarantined for the rest of their billing period to maximize cost efficiency (C3). The auto-scaling mechanism throughout has been assessed (C9). Results have shown reduced costs with improved response time and decreased SLA violations. In [109], Rahman, S., et al. have proposed an ML-based approach for VNF auto-scaling focused on dynamic traffic changes. The authors have presented an ML classifier that learns proactive measures from both past scaling decisions and temporal traffic behavioral patterns (C2). Results have demonstrated that the ML classifier improves the overall QoS and reduces costs (C3). SFC concepts and QoS metrics have been recently investigated in terms of auto-scaling. Lee, D., et al. [110] have proposed an auto-scaling method using RL for scale-in/out of multi-tier VNF instances (C2). The authors have defined the observation space based on SFC compositions while Service Level Objectives (SLOs) have been applied to design the reward function. Throughput (C9) and response time have been considered as SLOs. Results have shown an optimal number of VNF instances while minimizing SLO violation. In [111], Lin, T., et al. have studied the inclusion of network metrics into auto-scaling and scheduling mechanisms of cloud management systems. The authors have proposed an architecture that monitors the QoS (i.e. latency and bandwidth) of their clients. If QoS levels are violated, the auto-scaling system is activated, and new service instances are strategically deployed to meet QoS levels (C2). The authors have implemented a multi-tier and multi-tenant testbed,

incorporating heterogeneous (C8) devices (e.g. sensors, edge devices, DC locations) that span across geographically spread regions (C10). Efficient auto-scaling provides higher autonomy as shown by the adoption of ML-based methods.

### D. Performance Monitoring

Performance monitoring relates to the application's behavioral analysis. Monitoring or tracking applications allows service providers to fine-tune the application runtime performance while reducing allocation costs. This subsection surveys trends in performance monitoring research, discussing existing methods and tools that help to control the application execution and measure the impact of different infrastructures on their performance.

Containers, VMs and SDN applications have been a major topic in performance monitoring. In [112], Moradi, F., et al. have proposed an automated system for monitoring network performance of container-based applications called ConMon. It identifies newly instantiated containers and passively observes their traffic. Based on these observations, it configures and executes monitoring functions inside adjacent containers. Consequently, monitoring is isolated from the application and does not require instrumenting the image of the container or running additional processes inside it (C4). The feasibility and scalability (C2) of ConMon have been validated based on container performance and system resources. The authors have also assessed the impact of passive monitoring on the application's throughput (C9). In [113], Tangari, G., et al. have presented a decentralized approach for resource monitoring in SDN. Their proposal supports a wide range of measurement tasks and requirements regarding monitoring rates and information granularity levels. The authors have compared their decentralized system to a centralized approach based on a realistic use case, where a distributed management application coordinates a content distribution service in an ISP network. The trade-off between application reactivity/accuracy and monitoring scalability/overhead (C2) has been studied. The evaluation has considered requirements stemming from heterogeneous (C8) management applications. The average flow throughput (C9) has also been investigated in their monitoring scheme. In [114], Shah, S. Y., et al. have studied runtime dependencies among micro-services to detect anomalous behavior and meet SLAs. The authors have proposed a novel method based on Long-Short Term Memory (LSTM) recurrent NNs to find those dependencies in heterogeneous (C8) environments (i.e. public/private clouds). The authors focus on finding the strongest performance predictors, discovering temporal dependencies, and improving the accuracy of forecasting for a given performance metric. Results have proved its feasibility compared to existing literature methods, such as Granger causality and classical statistical time series models. Throughput (C9) levels have been assessed. In [115], Perdices, D., et al. have presented dPRISMA (distributed Passive Retrieval of Information, and Statistical Multipoint Analysis), a passive monitoring system generating statistical models for network measurements and raising alarms in case of anomalous behaviors. dPRISMA implements a distributed data gathering strategy, a promising approach to improve the scalability (C2) of monitoring systems. dPRISMA has been validated by correlating measurements retrieved from heterogeneous (C8) data sources (e.g. virtual environments, real-world data sets). SFC performance has also been analyzed in [116]. Sanz, I. J., et al. have developed a framework for the performance evaluation of SFC called SFCPerf. Their framework provides flexibility for experimenting with different NFs and virtualization infrastructures. Isolation (C4) has been addressed in the context of SFC since micro-service isolation has been applied through packet encapsulation. The authors adopt the Network Service Header (NSH) concept to perform packet forwarding in the service chain and isolate micro-services. To demonstrate the feasibility of SFCPerf, the authors consider a service chain constituted by virtual security (C5) functions. A service chain composed of an Intrusion Detection System (IDS) and a reactive firewall has been assessed concerning latency, throughput (C9) and the number of replied HTTP requests. Cited works show why monitoring tools are crucial for the orchestration life cycle. Without proper monitoring, performance decreases since adequate procedures are not employed.

### E. Caching

With the advent of modern wireless technologies and higher demand for multimedia services, new challenges have emerged for the support of multimedia content in next-generation networks. Caching at the edge is a promising approach to alleviate the burden on backhaul infrastructures, reduce data transmissions and minimize startup latency for multimedia delivery. The aim is to control data traffic and keep popular content at the edge close to end-users. Caching research has addressed mobility management in 5G contexts. Chen, M., et al. [117] have studied caching placement on small cell BSs and mobile devices by leveraging user mobility (C1), aiming to maximize the cache hit ratio. The authors have focused on delivering content while reducing energy consumption (C3). Results have shown that their approach improves cache hit ratio and energy efficiency compared to existing strategies. In [118], Tantayakul, K., et al. have studied a caching policy focused on mobility (C1) support in SDN networks. Two policies have been proposed: an on-off policy and an adaptive mechanism. Both methods have been analyzed for packet loss, channel occupancy, transmission time, throughput (C9) and bandwidth fairness. Both policies improve SDN mobility regarding packet loss. The authors state that the adaptive policy outperforms the on-off policy for latency-sensitive applications. The on-off policy provides long transmission times, though attaining enhanced channel occupancy, thus suitable for file transfer or media applications which download all contents before display. In [119], Zhang, K., et al. have proposed a cooperative edge caching architecture for 5G MEC. Their architecture focuses on mobility-aware hierarchical caching, where smart vehicles act as collaborative caching agents for sharing content operations with BSs (C1). Heterogeneous (C8) nodes (e.g. vehicles, mobile devices, BSs) have been considered as potential cache-enabled devices. In [120], Hao, Y., et al. have

introduced the concept of task caching for MEC, referring to the caching of completed task applications and their related data in edge-cloud infrastructures. The authors have studied the joint optimization of task caching and offloading through a MILP formulation. The authors have also proposed a heuristic algorithm that reduces energy costs compared to other schemes based on their simulations (C3). Heterogeneity (C8) has been satisfied through different task demands, several data sizes, and distinct service requirements. Cited works are promising approaches to support mobility in edge caching schemes.

Security has also been recently discussed. In [121], Xiao, L., et al. have studied attack models in MEC by focusing on both mobile offloading and caching procedures. The authors propose RL-based security solutions for safe offloading to edge nodes against jamming attacks. The aim is to improve the offloading quality, such as the Signal-to-Noise-plus-Interference Ratio (SINR) and Bit Error Rate (BER) of the signals received by edge nodes against jamming and interference to reduce energy consumption (C3). Their scheme reduces energy consumption and delay in mobile offloading while increasing the SINR of the signals received by edge nodes compared to benchmark schemes. The authors have also presented lightweight authentication and secure caching to preserve data privacy (C5). The authors state that security and data privacy are bottlenecks of MEC due to its heterogeneity (C8) in terms of devices (e.g. mobile, edge) and networks (e.g. physical, virtual). In [122], Cheng, F., et al. have proposed a novel caching scheme securing UAV-relayed wireless networks via jointly optimizing the UAV trajectory and time-scheduling. The authors suggest using UAVs as relay nodes for long-distance communications since they can be deployed on-demand because of their high mobility (C1) and agility. The authors state that energy consumption (C3), access delay, and throughput (C9) are essential factors for proper UAV operation. The authors propose exploiting energy harvesting techniques, such as solar energy, to provide sufficient energy supply to UAVs for delay-tolerant applications. The aim is to maximize throughput in multi-UAV networks while reducing their energy consumption. Simulations have proved the feasibility and efficiency of their scheme, showing the improved security of UAV relaying assisted networks (C5). Improving security in caching schemes will lead to its further acceptance in future networks.

### F. Summary

Section V presents trends on orchestration and management for next-generation networks. Resource allocation enables proper resource provisioning, while auto-scaling mechanisms ensure deployed services handle the current demand. Performance monitoring methods study application behavior, while caching research focuses on improving content delivery and reducing data transmissions, especially for fog/edge infrastructures. The literature review shows that resource allocation and caching address *Mobility* (C1) support, while Auto-scaling focuses on *Scalability* (C2). Reducing costs and managing multiple heterogeneous devices are open challenges of provisioning practices. Efficient provisioning is essential to support low latency service delivery. *Energy efficiency* (C3) is more noticeable in resource allocation and caching, while performance monitoring studies *Isolation* (C4). *Security* (C5) and data privacy is a major concern in caching schemes. *Resilience* (C6) is unexplored in all domains, while resource allocation addresses *Reliability* (C7). *Heterogeneity* (C8) is studied in all domains, while auto-scaling focuses on *Throughput* (C9) experiments. Auto-scaling systems capable of dealing with dynamic demands is a major future research topic. *Federation* (C10) concepts are still unexplored. All domains will play their role in the life-cycle management of containerized applications in future networks.

## VI. Integrating Machine Learning (ML) and Artificial Intelligence (AI): towards autonomous networks

### A. Overview

Over the past years, ML has become an interesting research field in the networking domain. ML methods have been adapted to traditional network problems. Due to the integration of ML / AI in network management, self-driving networks may emerge as a potential solution for inappropriate human intervention. This section reviews trends in ML/AI: DL, RL and FL. Table VII summarizes the reviewed research.

### B. Deep Learning (DL)

DL [142] is a subset of ML employing artificial NNs to learn complex problems from large amounts of data. DL methods learn without human supervision and from both structured and unlabeled data. Recently, DL has been applied to several domains, such as object and speech recognition, language translation, and computer vision. DL has also been recently adapted to traffic load and congestion prediction as in [126]. Tang, F., et al. have presented a DL-based algorithm for future traffic load and congestion prediction in an SDN-IoT network. Another DL algorithm has been proposed for channel assignment to avoid congestion in SDN-IoT. The authors consider a heterogeneous (C8) SDN-IoT network, where several devices can hardly cooperate, making it difficult to predict the traffic load sent by all devices. Simulations have demonstrated that their proposal outperforms conventional channel assignment algorithms regarding delay and throughput (C9).

DL has also been applied to SFC placement and routing. In [127], Pei, J., et al. have formulated the VNF selection and chaining problem as a Binary Integer Programming (BIP) model to minimize E2E delay. The authors have also presented a DL algorithm for the SFC routing problem. The evaluation has shown that DL obtains routing paths for SFC requests while achieving higher scalability (C2) compared to existing approaches. In [128], Jiang, F., et al. have proposed DL-based algorithms for resource scheduling in hybrid MEC networks. The authors have presented a large-scale path-loss fuzzy c-means algorithm to predict the optimal positions of Ground Vehicles (GVs) and UAVs that help with offloading tasks (C1). Their goal is to minimize the energy consumption (C3) of UEs by jointly optimizing the positions of GVs and UAVs, user association, and resource allocation. Simulations have

TABLE VII: Summary of the reviewed works in terms of ML / AI.

| Research Domain | Authors | Main focus | Year | Evaluation Criteria | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | C10 |
| Deep Learning (DL) (Sec. VI-B) | Tang, F., et al. [126] | Traffic load prediction | 2018 | × | × | × | × | × | × | × | ✓ | ✓ | × |
| | Pei, J., et al. [127] | SFC allocation | 2018 | × | ✓ | × | × | × | × | × | × | × | × |
| | Jiang, F., et al. [128] | Resource allocation for MEC | 2019 | ✓ | × | ✓ | × | × | × | × | ✓ | × | × |
| | Siasi, N., et al. [129] | SFC allocation | 2020 | × | ✓ | ✓ | × | × | × | × | ✓ | × | × |
| | Ali, Z., et al. [130] | Resource allocation for MEC | 2020 | ✓ | ✓ | ✓ | × | × | × | × | ✓ | × | × |
| Reinforcement Learning (RL) (Sec. VI-C) | Chen, L., et al. [131] | Auto-scaling | 2018 | × | ✓ | × | × | × | × | × | ✓ | ✓ | × |
| | Xu, Z., et al. [132] | Experience-driven networking | 2018 | × | × | × | × | × | × | × | × | ✓ | × |
| | Li, J., et al. [133] | Resource allocation for MEC | 2018 | × | × | ✓ | × | × | × | × | × | × | × |
| | Li, R., et al. [134] | Slicing allocation | 2018 | × | × | × | ✓ | × | × | × | ✓ | × | × |
| | Ye, H., et al. [135] | V2V communications | 2019 | ✓ | × | × | × | × | × | ✓ | ✓ | × | × |
| | Faraci, G., et al. [136] | 5G Slicing | 2020 | ✓ | × | ✓ | ✓ | × | × | × | × | × | × |
| Federated Learning (FL) (Sec. VI-D) | Tran, N. H., et al. [137] | Wireless Networks | 2019 | × | × | ✓ | × | ✓ | × | × | ✓ | × | × |
| | Chen, M., et al. [138] | Wireless Networks | 2020 | × | × | ✓ | × | × | × | × | × | × | × |
| | Qu, Y., et al. [139] | Privacy in Fog Computing | 2020 | × | ✓ | × | × | ✓ | × | × | × | ✓ | × |
| | Zhou, C., et al. [140] | Privacy in Fog Computing | 2020 | × | × | × | × | ✓ | × | × | × | × | × |
| | Kang, J., et al. [141] | Mobile Networks | 2020 | × | × | × | × | ✓ | × | ✓ | × | × | × |

shown that their framework achieves similar performance to heuristic methods while reducing CPU execution time for heterogeneous (C8) MEC networks. In [129], Siasi, N., et al. have presented a DL algorithm for SFC allocation in FC. Their scheme predicts the popularity of VNFs, mapping the most popular ones to High-Capacity Fog (HCF) nodes while linking unpopular NFs to Low-Capacity Fog (LCF) nodes. The authors remark that their strategy provides significant advantages since cached VNFs are available on nodes close to terminals. Caching alleviates congestion and saves resources by enhancing network scalability (C2) and capacity without increasing network cost. The authors have evaluated several performance metrics, including usage rate, network saturation, energy consumption (C3) and cost, based on a heterogeneous (C8) architecture composed of HCF and LCF nodes with different resource capacities. Lastly, Ali, Z., et al. [130] have presented a resource allocation algorithm for MEC called Power Migration Expand (PowMigExpand). Their algorithm assigns user requests to optimal servers and allocates optimal amounts of resources to UEs. Their approach migrates UE requests to new servers when needed due to user mobility (C1). A DL algorithm for user request allocation has also been proposed, considering the varying load of incoming requests. The authors have demonstrated the scalability (C2) of the PowMigExpand algorithm through simulations evaluating service rate, utility, and energy consumption (C3). Increased performance for a different number of ME servers and varying traffic has been obtained. The heterogeneity (C8) of user requests has been considered. Cited works highlight the multiple applications of DL algorithms, presenting them as an alternative to solve several challenges in the network management domain.

### C. Reinforcement Learning (RL)

In recent years, RL has become an important area in ML research [144]. Figure 7 represents a typical scenario in RL. RL has been applied to sequential decision-making. An agent learns to make better decisions from interacting with an environment, which represents the problem to solve. In the beginning, the agent knows nothing about the problem at hand and learns by performing actions in an environment. For each

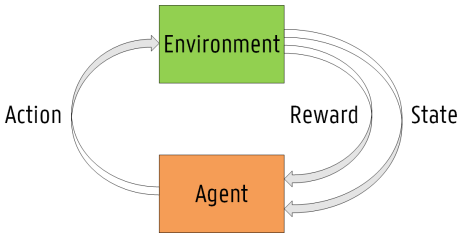



Fig. 7: The representation schema of most RL scenarios [143].

action taken, the agent receives a reward and a new observation that describes the new state of the environment. Depending on the goal and how the agent is performing the given task, the reward can be positive or negative. The agent learns to be successful by repeated interaction with the environment, by determining the inherent synergies between states, actions, and subsequent rewards. Ultimately, RL algorithms maximize the total reward an agent collects by experiencing multiple problem rounds. RL has been recently applied to Auto-scaling in [131]. Chen, L., et al. have addressed traffic optimization in a DC by applying Deep RL (DRL) algorithms. The authors have proposed a two-level RL system named AuTO to solve scalability (C2) problems in DCs. Their experiments have assessed homogeneous and heterogeneous (C8) traffic and measured flow throughput (C9) levels.

RL research has also focused on resource allocation and traffic control. In [132], Xu, Z., et al. have investigated the application of DL for model-free control in communication networks, focusing on traffic engineering. Simulations have shown that DL significantly reduces E2E delay and offers similar throughput (C9) to baseline methods. In [133], Li, J., et al. have proposed a DRL approach for computation offloading and resource allocation for wireless MEC. The authors have formulated the total cost of delay and energy consumption (C3) for all UEs as the optimization objective. Results have shown a significant reduction in the total cost compared to other baselines. In [134], Li, R., et al. have studied resource management for NS based on DRL. Simulations have demon-

strated the advantages of applying RL to radio resource slicing and priority-based slicing (C4). Heterogeneity (C8) has been addressed in distinct slices (e.g. video, URLLC). In [135], Ye, H., et al. have developed a decentralized resource allocation mechanism for V2V communications based on DRL. The high mobility (C1) of vehicles has been addressed in their scheme. A vehicle or a V2V link can decide which is the optimal sub-band and power level of transmission without requiring global information based on the decentralized nature of their approach. Latency and reliability (C7) requirements for V2V links have been discussed. Heterogeneous (C8) data (e.g. link interference, channel information) has been included in the observation space of the RL environment. The authors state that useful information can be extracted from heterogeneous data, simplifying optimal policy learning. Lastly, Faraci, G., et al. [136] have proposed the extension of 5G network slices with MEC UAVs. A system controller has been designed to handle job offloading to UAVs based on DRL. The authors aim to minimize power consumption (C3) and job loss while considering the mobility of UAVs (C1). An E2E logical network on top of physical infrastructures enables service isolation based on NS (C4). RL is still in the early stages but its applicability has already been proved, improving orchestration practices for low latency services.

### D. Federated Learning (FL)

FL [145] or collaborative learning is a recent ML technique that trains a model across multiple decentralized edge devices or servers holding local data samples without exchanging them. The edge devices download the current model from the central server and update the model based on their local data. Then, these trained local models are sent back to the central server, where they are aggregated (i.e. averaging weights) into a single global model that is sent back to all devices. The main benefit of FL is supporting decentralized learning since training data is kept locally without being transferred to central locations. Nevertheless, challenges persist in FL regarding efficient communications since edge devices need to share small portions of their training execution with the central server without transferring the complete data set. FL has been recently applied to wireless networks. In [137], Tran, N. H., et al. have formulated FL over wireless networks as an optimization problem. The authors have studied how the computation and communication latency of UEs affect their energy consumption (C3) and both learning time and accuracy. The authors remark on the benefits of FL concerning data privacy since local data is not shared (C5). Numerical results have quantified the impact of UE heterogeneity (C8) on the system cost. In [138], Chen, M., et al. have also applied FL over wireless networks. In their scheme, wireless users train their local model using their data samples and transmit the trained model to a BS. The authors have studied the joint problem of wireless resource allocation and user selection for running FL algorithms. Their formulation aims to minimize the training loss while meeting delay and energy consumption (C3) requirements.

Privacy concerns have also been recently addressed. In [139]. Qu, Y., et al. have proposed a BC-enabled FL scheme for privacy preservation in FC. FL enables decentralized learning while BC allows end devices to exchange model updates based on a consensus mechanism without any centralized authority. Their FL approach achieves learning convergence when data sizes are moderate, proving its scalability (C2). Their scheme has been assessed regarding privacy protection, efficiency, and resistance to poisoning attacks (C5). In the evaluation, the authors consider a throughput (C9) model based on the Shannon capacity with a certain loss. In [140], Zhou, C., et al. have proposed a privacy-preserving FL scheme for FC. In their approach, fog nodes collect data from IoT devices to perform learning tasks, improving training efficiency and accuracy. Differential privacy mechanisms and security aggregation methods have been considered to protect device data and protect devices from collusion attacks (C5). Mobile networks have also been discussed in [141]. Kang, J., et al. have presented the concept of reputation as a metric to discover trustworthy nodes in FL. The authors have leveraged consortium BC for achieving efficient node management without repudiation and tampering (C5). Results have shown improved reliability (C7) of FL tasks over mobile networks. Based on reviewed research, FL has become an adequate solution for user privacy preservation.

### E. Summary

Section VI presents trends in ML research being adopted in network management. DL has been applied to networking problems based on large data sets, while RL has studied how to perform tasks by interacting with an environment without being given any information beforehand. In contrast, FL enables decentralized learning where data remains local, and each device runs its copy of the model. The literature review shows that RL and DL focus on *Mobility* (C1) and *Scalability* (C2). *Energy efficiency* (C3) is addressed in all domains, while *Isolation* (C4) is discussed in RL. FL focuses on *Privacy* (C5) preservation. Keeping data private is a major challenge in future networks. Users will share their information (e.g. mobility pattern, application preferences) with service providers to receive higher QoE. Ensuring privacy and security while improving network performance is a future research topic. *Resilience* (C6) is unexplored in all domains, while RL and FL address *Reliability* (C7). *Heterogeneity* (C8) is studied in all domains, while *Throughput* (C9) is more noticeable in RL. *Federation* (C10) is unexplored. Combining these ML trends can lead to fully automated networks with minimum human intervention. Self-configuration and self-repairing features will strongly impact the performance of low latency services.

## VII. Security and privacy mechanisms for cloud-native infrastructures

### A. Overview

Security and Privacy are crucial enablers of emerging use cases in future networks. Without proper authentication/authorization mechanisms and data privacy solutions, service providers will not benefit from improved technologies since users will only subscribe to services that protect their privacy and data. This section discusses trends in security

TABLE VIII: Summary of the reviewed works in terms of Security and Privacy.

| Research Domain | Authors | Main focus | Year | Evaluation Criteria | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | C10 |
| BlockChain (BC) (Sec. VII-B) | Dorri, A., et al. [146] | IoT | 2017 | × | ✓ | ✓ | × | ✓ | × | × | × | × | × |
| | Zamani, M., et al. [147] | BC resiliency | 2018 | × | ✓ | × | × | ✓ | ✓ | × | × | ✓ | × |
| | Alvarenga, I. D., et al. [148] | Management of VNFs | 2018 | × | × | × | × | ✓ | ✓ | × | × | × | ✓ |
| | Dinh, T. T. A., et al. [149] | BC framework | 2018 | × | ✓ | × | × | ✓ | ✓ | × | × | ✓ | × |
| | Zheng, W., et al. [150] | BC platform | 2019 | × | ✓ | × | × | ✓ | × | ✓ | × | × | × |
| | Qiu, Z., et al. [151] | IoT | 2020 | × | ✓ | ✓ | × | ✓ | × | ✓ | × | ✓ | × |
| CyberSecurity (CS) (Sec. VII-C) | Varga, P., et al. [152] | SDN | 2017 | × | ✓ | × | × | ✓ | × | × | × | ✓ | × |
| | Diro, A. A., et al. [153] | Fog Computing | 2017 | × | ✓ | × | × | ✓ | × | × | × | × | × |
| | Sohal, A. S., et al. [154] | Fog Computing | 2018 | × | × | × | × | ✓ | × | × | × | × | × |
| | Dzeparoska, K., et al. [155] | Architecture | 2018 | × | ✓ | × | × | ✓ | × | × | ✓ | ✓ | × |
| | Joshi, K. D., et al. [156] | Privacy in SDN | 2019 | × | ✓ | × | × | ✓ | × | × | × | × | ✓ |
| Trusted Computing (TC) (Sec. VII-D) | Aga, S., et al. [157] | Smart memory | 2017 | × | × | ✓ | ✓ | ✓ | × | × | × | × | × |
| | Gjerdrum, A. T., et al. [158] | TC performance | 2017 | × | × | × | ✓ | ✓ | × | × | × | × | × |
| | Coughlin, M., et al. [159] | TC for secured NFV | 2017 | × | × | × | ✓ | ✓ | × | × | × | ✓ | × |
| | Yin, H., et al. [160] | Decentralized TC | 2018 | × | × | × | × | ✓ | × | × | × | × | × |

research: BC, CS and TC. Table VIII summarizes the reviewed works.

### *B. BlockChain (BC)*

BC [161] is a decentralized digital ledger of transactions distributed across a network of nodes based on P2P topologies. A list of transactions is called a block, which is then linked with other blocks through cryptographic hashes. Nodes keep a time-stamped series of immutable data records without a central authority. By design, BC is resilient to data modification since once a block is recorded, data modifications cannot happen without altering all subsequent blocks. BC is a potential enabler of efficient and secure sharing of information in next-generation networks due to its decentralized nature. BC has already been studied for IoT contexts in [146]. Dorri, A., et al. have proposed a lightweight BC-based architecture for IoT, eliminating overheads of classic BC methods. The authors adopt a hierarchical structure employing a centralized immutable ledger to reduce overhead, increase network scalability (C2), and optimize resource consumption (C3). Evaluations have demonstrated the robustness of their architecture against several attacks (C5).

BC resilience has also been studied. In [147], Zamani, M., et al. have presented RapidChain, a sharding-based public BC protocol, and assessed its scalability (C2) and security (C5). RapidChain is resilient (C6) to Byzantine faults from up to a 1/3 fraction of its participants. RapidChain employs an optimal intra-committee consensus algorithm achieving high throughput (C9) levels via block pipelining, a novel gossiping protocol for large blocks. In [148], Alvarenga, I. D., et al. have proposed a BC architecture for secure management, configuration, and migration of VNFs. Their approach ensures the integrity and consistency of VNF information while keeping the anonymity of tenants and configuration information (C5). The authors adopt a consensus protocol that validates every transaction before storing it in a block to provide resiliency against faulty systems and collusion attacks (C6). The authors consider a federated consensus model where participants are known, and their asymmetric key pair is certified by a third-party, previously agreed upon by the federation members (C10). In [149], Dinh, T. T. A., et al. have presented BLOCKBENCH, a benchmarking framework for understanding the performance of private BCs against data processing workloads. The authors have evaluated three major BC systems (i.e. Ethereum, Parity, and Hyperledger Fabric) regarding scalability (C2), fault tolerance and security (C5), throughput (C9), and latency. The authors conclude that Ethereum and Parity are more resilient (C6) to node failures but more vulnerable to security attacks that fork the BC. Cited works show that BC architectures can enable adequate resilience for next-generation networks.

BC architectures also acknowledge the importance of reliability. In [150], Zheng, W., et al. have developed a BC-as-a-service platform called NutBaaS, in which BC services (e.g. smart contracts, system monitoring) are enabled over cloud infrastructures. To improve NutBaaS flexibility and scalability (C2), the authors have adopted Kubernetes to automatically deploy container-based services. The authors state that NutBaaS helps developers to detect security vulnerabilities by providing smart contract services, thus avoiding economic losses (C5). The authors have also discussed the reliability (C7) of their NutBaaS platform. In [151], Qiu, Z., et al. have proposed a Dual Vote Confirmation based Consensus (DVCC) mechanism for the integration of BC and IoT. Firstly, the authors have adopted a role division approach to handle limited resources in IoT devices. Then, a dual confirmation algorithm has been designed to improve the security and fairness of the consensus. A voting method has also been presented to enhance delay performance (C2) and avoid energy waste (C3). Results have shown that DVCC guarantees the security (C5) and fairness of the consensus while achieving notable reliability (C7) and throughput (C9). Both works propose adequate BC-based architectures solving current reliability issues while providing efficient security measures.

### *C. CyberSecurity (CS)*

CS [162] consists of protecting computer systems, networks and data from malicious attacks or threats. Organizations and enterprises apply security methods to protect against unauthorized access to their infrastructures. Hardware-accelerated security practices have been studied in [152]. Varga, P., et al.

[152] have introduced an automated method to speed up the reaction lag in SDN-based DCs via FPGA-based processing units. The authors aim to enhance the security (C5) of DCNs and large clouds with a new FPGA-accelerated NFV. The authors evaluate their approach in a DCN with a new security application that detects and mitigates real-world Distributed Denial of Service (DDoS) attacks, with lags from 430 $\mu$s up to 3 ms - several orders of magnitude faster than traditional approaches (C2). The authors state that their FPGA-based closed loop achieves higher throughput (C9) levels and lower latency than conventional methods. Security aspects in Fog Computing have also been addressed recently. In [153]. Diro, A. A., et al. have proposed an FC-based publish-subscribe lightweight protocol for IoT and assessed its scalability (C2) and security (C5). Their scheme provides higher scalability and less overhead than traditional methods while guaranteeing similar levels of security. In [154], Sohal, A. S., et al. have proposed a CS framework to identify malicious edge devices in FC. The authors adopt a two-stage hidden Markov model to categorize edge devices while a Virtual Honeypot Device (VHD) stores information of all identified malicious devices to assist the system, securing it from future attacks (C5).

Security for SDN has been studied in [155]. Dzeparoska, K., et al. have proposed a hierarchical, logically centralized architecture, expressing security policies through Autonomous Systems (AS) as intents. The authors state that SD Internet Exchange Points (SDXs) provide flexible and programmable control over wide-area network traffic delivery. SDX collaboration has addressed scalability (C2) challenges since security (C5) intents are compiled and installed at the available SDX closest to the malicious source, effectively protecting against DDoS attacks. A heterogeneous (C8) topology (i.e. SDXs, SDN controllers) has assessed the throughput (C9) of their approach. In [156], Joshi, K. D., et al. have presented PRIME-Q, a privacy-aware E2E QoS framework in Multi-domain SDN. A privacy (C5) index has been defined to quantify the privacy level of E2E QoS coordination. Simulations have assessed the scalability (C2) and operational overhead of PRIME-Q in large multi-domain networks (C10). These works are appropriate answers to attacks or threats in the coming years.

### D. Trusted Computing (TC)

TC [163] refers to technologies and proposals for solving security problems through enhancing hardware or modifying software to secure cloud computing and virtualized systems. In recent years, research has focused on improving hardware performance. In [157], Aga, S., et al. have proposed InvisiMem, a memory approach expanding the trust base to include the logic layer in the smart memory to cryptographic primitives. It allows the secure host processor to send encrypted addresses over the untrusted memory bus. InvisiMem significantly improves memory space, energy (C3), and overhead. The authors also state that InvisiMem properly supports enclave isolation (C4). The authors conclude that InvisiMem ensures similar security (C5) to Oblivious RAM (ORAM)-based solutions. In [158], Gjerdrum, A. T., et al. have studied the performance characteristics of SGX technology to understand how it could enforce privacy policies in cloud-hosted Software-as-a-Service (SaaS) architectures (C5). The authors also state that developers should pre-provision enclaves in a disposable pool of resources to prevent reuse between isolation (C4) domains if before-the-fact usage of enclaves is accurately predicted. In [159], Coughlin, M., et al. have studied the application of TC to circumvent limitations on encrypted data. Encrypted data introduces an overhead while providing a limited set of operations since encrypted schemes are not completely homomorphic. The authors have studied whether SGX technologies can perform secure packet processing since SGX provides proper software isolation (C4) and attestation (C5). A real NFV environment has been evaluated, including throughput (C9) experiments. In [160], Yin, H., et al. have presented HyperNet, a decentralized TC and networking paradigm to address control loss over data. The decentralized trusted connection is based on BC smart contracts enabling secure digital object management and an identifier-driven routing mechanism (C5). The authors remark that HyperNet handles data sovereignty and helps to build a Universal Data object Identifier (UDI)-driven network capable of indexing and routing data. Cited works show that increasing hardware trust is the first step towards efficient security in future networks.

### E. Summary

Section VII reviews novel security mechanisms enabling data privacy and secure transmissions in cloud-native systems. BC enables a decentralized trust system, while CS implements security practices to defend cloud systems from potential threats. TC focuses on trust and software isolation through hardware-based protection. The literature review shows that *Mobility* (C1) is unaddressed, while BC and CS address *Scalability* (C2) issues arising when adopting novel security mechanisms. *Energy efficiency* (C3) is more noticeable in BC, while TC focuses on *Isolation* (C4). As expected, these domains mainly discuss *Security* and *Privacy* (C5). BC also addresses *Resilience* (C6) and *Reliability* (C7), while *Heterogeneity* (C8) is studied in CS. BC and CS also address *Throughput* (C9) and *Federation* (C10). Securing future networks requires all three domains. A decentralized architecture fully supports low latency services, as long as efficient security against attacks and high trust when executing NFs in hardware is guaranteed.

## VIII. Emerging Applications for next-generation networks

### A. Overview

The emerging cloud-native infrastructures and novel technologies lead to new use cases requiring even more stringent requirements (e.g. higher reliability, lower latency). This section presents research focused on four emerging use cases: Smart Cities, Self-driving cars, XR, and IIoT. Table IX summarizes the reviewed works.

TABLE IX: Summary of the reviewed works in terms of Application.

| Research Domain | Authors | Main focus | Year | Evaluation Criteria | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | C1 | C2 | C3 | C4 | C5 | C6 | C7 | C8 | C9 | C10 |
| Smart Cities (Sec. VIII-B) | Montori, F., et al. [164] | Architecture | 2017 | × | × | × | × | × | × | ✓ | ✓ | × | × |
| | Cheng, B., et al. [165] | Fog Computing | 2017 | × | ✓ | ✓ | × | × | × | ✓ | ✓ | ✓ | ✓ |
| | Ejaz, W., et al. [166] | Energy management | 2017 | × | × | ✓ | × | × | × | × | ✓ | × | × |
| | Santos, J., et al. [167] | Resource allocation | 2018 | × | ✓ | × | × | × | × | × | ✓ | × | × |
| | Liu, Y., et al. [168] | Energy management | 2019 | × | × | ✓ | × | × | × | × | × | × | × |
| | Aloqaily, M., et al. [169] | Security | 2019 | × | × | ✓ | × | ✓ | × | × | × | × | × |
| | Santos, J., et al. [170] | Resource allocation | 2020 | ✓ | ✓ | ✓ | ✓ | × | × | × | ✓ | × | × |
| Self-driving cars (Sec. VIII-C) | Maqueda, A. I., et al. [171] | Steering Prediction | 2018 | × | × | × | × | × | × | ✓ | × | × | × |
| | Chen, S., et al. [172] | Testing platform | 2019 | × | × | × | × | × | × | ✓ | × | × | × |
| | Chernikova, A., et al. [173] | Security | 2019 | × | × | × | × | ✓ | × | × | × | × | × |
| | Ndikumana, A., et al. [174] | Caching for MEC | 2020 | ✓ | × | × | × | × | × | × | ✓ | ✓ | × |
| Extended Reality (XR) (Sec. VIII-D) | Liu, Y., et al. [175] | MEC-assisted VR | 2018 | × | × | ✓ | × | × | × | × | × | × | × |
| | Doumanoglou, A., et al. [176] | Quality of Experience | 2018 | × | × | × | × | × | × | ✓ | × | ✓ | × |
| | van der Hooft, J., et al. [177] | Adaptive VR services | 2019 | × | × | × | × | × | × | × | × | ✓ | × |
| | van der Hooft, J., et al. [178] | Point cloud compression | 2019 | × | × | × | × | × | ✓ | × | × | ✓ | × |
| Industrial IoT (IIoT) (Sec. VIII-E) | Zhou, L., et al. [179] | Data processing | 2017 | × | × | ✓ | × | × | × | × | ✓ | × | × |
| | Cheng, J., et al. [180] | 5G smart manufacturing | 2018 | × | × | ✓ | × | ✓ | × | ✓ | ✓ | × | × |
| | Luvisotto, M., et al. [181] | indoor IIoT | 2018 | × | × | ✓ | × | × | × | ✓ | × | × | × |
| | Hiller, J., et al. [182] | Low Latency | 2018 | × | × | ✓ | × | ✓ | × | × | × | × | × |
| | Yang, H., et al. [183] | Low Latency | 2019 | ✓ | × | ✓ | × | × | × | ✓ | ✓ | ✓ | × |
| | Niya, S. R., et al. [184] | Blockchain architecture | 2020 | × | × | ✓ | × | ✓ | × | ✓ | ✓ | ✓ | × |
| | Kumar, T., et al. [185] | Blockchain architecture | 2020 | × | ✓ | ✓ | × | ✓ | × | ✓ | ✓ | × | × |

### *B. Smart Cities*

A Smart City [186] applies technology to improve the performance of its urban services by transforming simple objects into smart connected devices. Sensors are spread around the city to collect data and then insights are extracted from these data to improve urban operations and services. Several domains of urban life are affected by these types of applications, such as waste management, environmental monitoring, urban mobility, and healthcare. Research has been studying suitable architectures for IoT services. In [164], Montori, F., et al. have introduced an architecture to handle data sources in IoT. The authors have also proposed crowdsensing management functionalities for environmental data, addressing reliability (C7). Their approach focuses on the integration of heterogeneous (C8) data sources. In [165], Cheng, B., et al. have designed and implemented an FC framework for Smart Cities called FogFlow. Message propagation latency, scalability (C2) and energy consumption (C3) have been assessed based on an anomaly detection use case. The authors state that FogFlow is a reliable (C7) option to handle multiple IoT brokers in a Smart City environment. The heterogeneity (C8) of IoT has been considered through different device profiles (e.g. temperature sensor, alarm) and distinct contexts (e.g. data processing, service management). Throughput (C9) has also been evaluated. The authors also propose a federated broker to exchange context information with other federated brokers in a multi-domain environment (C10). In [166], Ejaz, W., et al. have discussed efficient energy management in Smart Cities. The authors have addressed energy harvesting in a Smart City to extend the lifetime of low-powered devices (C3). The heterogeneous (C8) nature of IoT has also been considered. In [167], the City of Things (CoT) framework has been presented for data collection and analysis and automated resource provisioning in Smart Cities. The framework has been evaluated on an air quality monitoring use case by deploying air quality sensors in cars. CoT is a flexible and scalable (C2) approach for the Smart City ecosystem. Heterogeneity (C8) has been discussed through different devices and distinct functionalities (e.g. ML algorithms, data operations). In [168], Liu, Y., et al. have designed an IoT-based energy management system with DRL. An efficient energy scheduling method has been presented to manage the uncertainty of energy supply and demand in a Smart City (C3). All cited architectures are appropriate to deal with massive numbers of connected devices and the stringent requirements of IoT.

Security mechanisms have also been studied in [169]. Aloqaily, M., et al. have introduced a cloud framework for connected vehicles focused on intrusion detection mechanisms against security attacks while meeting user QoS levels. The authors have discussed the future role of connected cars in the electric power grid (C3), by suggesting the stabilization of the power grid through wirelessly charging batteries. Simulations have shown that their approach significantly mitigates real attacks (C5). Lastly, in [170], a MILP formulation for IoT service allocation has been proposed, which considers SFC concepts, different Low Power Wide Area Network (LPWAN) technologies, and multiple optimization objectives. Mobility (C1) has been addressed since users move around in the network, and the MILP model considers the impact of service migrations. The scalability (C2) of the MILP formulation has been assessed for several optimization objectives, including energy efficiency (C3) and E2E latency. The concept of NS has also been included in the model by adding different slices to multiple applications (C4). Heterogeneity (C8) has been satisfied through different service requirements (e.g. CPU, RAM) and distinct LPWAN technologies (i.e. IEEE 802.11ah and LoRaWAN).

### C. Self-driving cars

Self-driving or autonomous vehicles [187] are cars or trucks in which little to no human intervention is needed. These vehicles sense the environment and drive safely through software-based control decisions. Self-driving cars are getting significant attention in several research domains. In [171], Maqueda, A. I., et al. have applied DL for steering prediction in self-driving cars. The authors aim to investigate whether DL algorithms provide robust steering prediction based on data samples from event cameras. Their approach produces reliable (C7) steering angle predictions in challenging situations (e.g. poor lighting, fast motion). In [172], Chen, S., et al. have proposed an integrated simulation system for self-driving vehicles. The approach consists of four main components: the vehicle kinematic model simulation, the multi-sensor simulation, the environment simulation, and the Electronic Control Unit (ECU) hardware. Several experiments have validated their platform, showing its performance and reliability (C7).

Security in self-driving cars has also been recently studied. In [173]. Chernikova, A., et al. have proved that evasion attacks are a threat to steering angle predictions in autonomous vehicles (C5). The authors have stated that these attacks threaten the safe applicability of DL in autonomous driving. In [174], Ndikumana, A., et al. have proposed infotainment caching in self-driving cars, in which caching decisions are made based on passenger features obtained through DL. DL predicts cached contents in autonomous cars or MEC servers attached to roadside units. The authors state that lower delays for downloading operations are achieved if autonomous cars select available MEC servers en route since self-driving cars are delay-sensitive due to their high mobility (C1). In their evaluation, heterogeneous contents (i.e. movies with different demands and user recommendations) have been considered and throughput (C9) levels have been assessed.

### D. Extended Reality (XR)

XR [188] refers to all real and virtual environments generated via computers or even wearables that blend virtual and physical worlds to create fully immersive experiences. In AR, objects are overlaid into the real world by enhancing the user experience through AR glasses. XR is one of the emerging use cases for future networks that will completely revolutionize multimedia service delivery. MEC-assisted VR has been studied in [175]. Liu, Y., et al. have proposed a Panoramic VR Video (PVRV) streaming system designed for millimeter-wave (mmWave) mobile networks in combination with MEC. The authors remark that adopting MEC servers can improve wireless bandwidth utilization and UE energy efficiency (C3). Simulations have shown that their PVRV system improves energy efficiency and the quality of received viewport over conventional methods. QoE of VR streaming services has also been addressed in [176]. Doumanoglou, A., et al. have studied the QoE of real-time 3D media content streamed to VR headsets for entertainment purposes. The aim is to embed real users within virtual environments of interactive games to provide a fully immersive experience. The evaluation has considered multiple users under varying network conditions to assess the overall QoE concerning a range of visual quality and latency parameters. The trade-off between latency introduced by a reliable transport protocol (i.e. TCP) versus frame loss rate has been assessed (C7). In their experiments, lag issues occurred when using TCP, though the authors claim that a buffering mechanism potentially mitigates these issues. Throughput (C9) assumptions for TCP and User Datagram Protocol (UDP) have been included in their scheme.

VR adaptive streaming has also been investigated. In [177], three research challenges in immersive streaming have been tackled: viewport prediction, tile-based rate adaptation and application layer optimizations. A content-agnostic viewport prediction scheme based on spherical walks alongside a novel rate adaptation heuristic for tile-based video has been proposed. The advantages of using HTTP/2 server push have been studied since it significantly improves viewport prediction, video quality, and throughput (C9) compared to existing methods. In [178], HTTP adaptive streaming of VR content through Point Cloud Compression (PCC) has been studied. Several rate adaptation heuristics have been presented to decide the most appropriate quality representation of each VR object. The trade-off between accurate prediction and resilience (C6) to playout freezes has been discussed. Increasing the buffer size results in lower interactivity, prediction accuracy, and video quality. However, a larger buffer results in higher resilience to playout freezes, a crucial factor for over-the-top video streaming. Throughput (C9) traces have been considered in the evaluation. Cited works aim to provide efficient mechanisms towards maximizing users QoE.

### E. Industrial IoT (IIoT)

IIoT [189] refers to the extension of IoT towards the industrial sector. Sensors interconnect with manufacturing processes and robots easing data collection to improve productivity and efficiency of industrial processes. Energy consumption has been an important subject in IIoT. In [179], Zhou, L., et al. have studied the performance of different computing methods in IIoT by analyzing the relationship between data processing and energy consumption (C3). Heterogeneity (C8) has been discussed in the consumption estimations of data transmissions from multiple sources. In [180], Cheng, J., et al. have proposed a 5G-based IIoT architecture while describing different manufacturing functionalities for three use cases: eMBB, mMTC and URLLC. The authors discuss the importance of low-cost and low-energy consumption in 5G (C3). Data privacy and security (C5) aspects have been considered alongside reliability (C7) and heterogeneity (C8). The authors remark that real-time monitoring in IIoT requires a packet loss rate lower than $1 \times 10^{-12}$. Current 4G technologies cannot meet these requirements, while 5G wireless communications are promising for URLLC services. In [181], Luvisotto, M., et al. have assessed the performance of LoRaWAN for typical IIoT use cases, such as indoor industrial monitoring. The authors discuss how particular parameters can be adapted to increase the performance of their industrial scenario. Simulations have shown that LoRaWAN is a viable alternative for these applica-

tions since it ensures low energy consumption (C3) and high reliability (C7).

Low latency communications have also been recently addressed. In [182], Hiller, J., et al. have studied secure low latency communications in IIoT. The authors propose antedated encryption and fast data authentication with templates, securing low-powered devices (C3). The authors state that implementing security (C5) measures on constrained IoT devices adds further latency overhead. Security processes execution time increases the perceived latency. Evaluations have quantified the overhead of their approach focused on latency and energy consumption, showing that IIoT latency requirements are met through antedated encryption and fast data authentication. In [183], Yang, H., et al. have proposed a heterogeneous Radio Frequency (RF) / Visible Light Communication (VLC) architecture to satisfy different QoS requirements for URLLC devices. A resource management approach has also been formulated as a Markov Decision Process (MDP), followed by an RL algorithm that learns the optimal policy. The authors consider user mobility (C1) in the RL policy strategy. The heterogeneous (C8) RF/VLC architecture has been evaluated concerning energy consumption per device (C3), reliability (C7) and throughput (C9). Results have shown that their approach enables energy-efficient communications, satisfies URLLC requirements (i.e. high reliability, low latency), and ensures high data rates at different scenarios in IIoT. Low latency in IIoT is essential since high delays can have major repercussions (e.g. damaged machines, injured employees). Both works show promising results to guarantee low latency communications in industrial scenarios.

BC architectures have been proposed for IIoT. In [184], Niya, S. R., et al. have proposed a BC-agnostic architecture called BIIT for the IIoT. BIIT aims to reduce computational overhead and enhance energy efficiency (C3). The authors intend to offer higher levels of trust (C5), transparency, and data reliability (C7) by leveraging BC. Heterogeneity (C8) has been discussed regarding data collection. The authors have also studied performance issues in their architecture in combination with LoRa, focusing on throughput (C9) concerns. The authors maximize throughput while simultaneously providing data integrity through cryptographic signatures. The authors suggest performing large-scale experiments with BIIT as future work, validating the scalability of the system in LoRa and cellular networks. In [185], Kumar, T., et al. have proposed BlockEdge, a framework combining edge computing with BC to address the stringent requirements introduced by IIoT. The feasibility of BlockEdge has been assessed regarding scalability (C2), latency, power consumption (C3) and network usage compared to non-BC approaches. Results prove that BlockEdge provides decentralized trust and security (C5) management in IIoT without compromising system performance and resource efficiency. In contrast, the system reliability (C7) is improved. The Heterogeneity (C8) of IIoT environments has also been addressed. Cited works show that BC supports reliable communications in IIoT.

### F. Summary

Section VIII reviews emerging applications for future networks. Self-driving cars address *Mobility* (C1) concerns, while Smart Cities and IIoT focus on *Scalable* (C2) and *Energy-efficient* (C3) platforms, acknowledging the importance of low power consumption in these environments. Few works address *Isolation* (C4), while *Security* (C5) is a major concern in self-driving cars and IIoT. *Resilience* (C6) is mostly unexplored, while *Reliability* (C7) concerns are addressed in self-driving cars and IIoT. Smart Cities and IIoT address their *Heterogeneous* (C8) nature due to IoT, while XR discusses *Throughput* (C9) since these applications require high bandwidth data rates, as well as low latency as part of QoE expectations. *Federation* (C10) is mostly unaddressed. All these applications are pushing towards paradigm shifts in architecture, communication, orchestration and security. Addressing current hurdles will help next-generation networks to fully support these applications and deliver low E2E latency. The next section discusses open challenges and future directions.

## IX. Open Challenges & Future Directions

To tackle open challenges in low latency service delivery, several works addressing different domains need to be combined to meet all criteria shown in previous summary Tables (i.e. IV to IX). This section discusses challenges not yet fully addressed by literature and highlights future directions. Table X summarizes the reviewed works aggregated per evaluation criteria. By combining research from different domains, readers can easily consult works addressing a particular criterion of their interest.

*Mobility* (C1) has been addressed by several works. Nevertheless, no generic approach has been proposed to ensure service operation when end-users or devices are moving. MEC (e.g. [37], [38]) and FC (e.g. [45], [46]) address mobility focused on mobile devices and IoT, respectively. A viable alternative is merging both domains and provide an integrated solution for the mobility support of different devices (e.g. end-users, IoT sensors and vehicles). Integrating the mobile and the cloud domain is a research direction that will certainly help address future mobility requirements. NS [69], [71] and IBN [84] have also been proposed to handle user mobility and usage patterns. Mobility and access functionalities can be deployed and managed independently. Future research will focus on NS and IBN to offer mobility alongside high degrees of flexibility and customization. Resource allocation and caching have also addressed mobility requirements. However, most works focus only on specific use cases, such as D2D networks [100] or MEC [119]. ML has also been studied for mobility management in V2V communications [135] or MEC (e.g. [128], [130]). These methods aim to autonomously manage user mobility, but no clear approach has been yet applied in practice. Emerging use cases such as autonomous vehicles, IIoT, and in particular, Smart City services would definitively benefit from enhanced mobility support in future cloud-native infrastructures.

*Scalability* (C2) has been tackled in the revised literature, starting from different points of view. Novel architectures

TABLE X: Summary of revised works on low latency service delivery.

| Evaluation Criteria | Relevant reviewed works | Number of Publications |
|---|---|---|
| Mobility (C1) | [37], [38], [40], [42], [45], [46], [49], [50], [69], [71], [84], [100], [103], [107], [117], [118], [119], [122], [128], [130], [135], [136], [170], [174], [183] | 25 (20.8%) |
| Scalability (C2) | [43], [45], [46], [49], [48], [47], [52], [53], [54], [55], [56], [57], [58], [71], [73], [75], [78], [79], [80], [81], [85], [86], [87], [88] [89], [90], [104], [108], [109], [110], [111], [112], [113], [115], [127], [129], [130], [131], [139], [146], [147], [149], [150] [151], [152], [153], [155], [156], [165], [167], [170], [185] | 52 (43.3%) |
| Energy Efficiency (C3) | [39], [49], [50], [56], [57], [76], [77], [85], [101], [102], [103], [104], [105], [106], [107], [108], [109], [117], [120], [121], [122], [128], [129], [130], [133], [136], [137], [138], [146], [151], [157], [165], [166], [168], [169], [170], [175], [179], [180], [181], [182], [183], [184], [185] | 44 (36.6%) |
| Isolation (C4) | [38], [44], [48], [52], [55], [69], [70], [71], [72] [75], [79], [81], [83], [104], [112], [116], [134], [136], [157], [158], [159], [170] | 22 (18.3%) |
| Security & Privacy (C5) | [38], [43], [44], [46], [47], [50], [51], [52], [53], [71], [79], [82], [84], [116], [121], [122], [137], [139], [140], [141], [146], [147], [148], [149], [150], [151], [152], [153], [154], [155], [156], [157], [158], [159], [160], [169], [173], [180], [182], [184], [185], | 41 (34.2%) |
| Resilience (C6) | [47], [88], [90], [147], [148], [149], [178] | 7 (5.8%) |
| Reliability (C7) | [39], [40], [41], [42], [46], [47], [49], [50], [69], [70], [72], [73], [74], [78], [80], [88], [90], [100], [106], [107], [135], [141], [150], [151], [164], [165], [171], [172], [176], [180], [181], [183], [184], [185] | 34 (28.3%) |
| Heterogeneity (C8) | [40], [42], [43], [45], [46], [47], [48], [49] [50], [53], [54], [55], [56], [69], [70], [72], [75], [78], [79], [80], [81], [83], [86], [87], [100], [101], [102], [103], [105], [107], [111], [113], [114], [115], [119], [120], [121], [126], [128], [129], [130], [131], [134], [135] [137], [155], [164], [165], [166], [167], [170], [174], [179], [180], [183], [184], [185] | 57 (47.5%) |
| Throughput (C9) | [37], [38] [39], [40], [41], [45], [46], [47], [49], [51], [55], [56], [57], [58], [70], [69], [72], [73], [74] [76], [77], [75], [79], [82], [83], [85], [88], [100], [106], [105], [108], [110], [112], [113], [114], [116], [118], [122], [126], [131], [132], [139], [147], [149], [151], [152], [155], [159], [165], [174], [176], [177], [178], [183], [184] | 55 (45.8%) |
| Federation (C10) | [42], [46], [77], [80], [81], [86], [111], [148], [156], [165] | 10 (8.3%) |

(e.g. [43], [48]) have been designed to focus on application scalability or in supporting a high number of connected devices. Hardware-accelerated platforms have also been studied for performance improvement regarding NFV reconfiguration [55], memory access [57] and data processing pipelines [58]. Their purpose is to enhance the performance and scalability of hardware-based platforms running softwarized NFs. Several works have addressed scalability issues in SFC (e.g. [73], [75]), IBN [80], [81] and SR (e.g. [87], [88]). All these networking paradigms aim to provide higher levels of flexibility and reconfigurability to current networks, but their impact on network scalability needs to be acknowledged. Future research will focus on implementing novel mechanisms in practical testbeds and assess their scalability and overhead concerning execution time and resource usage. Auto-scaling (e.g. [108], [109]) and performance monitoring systems (e.g. [112], [113]) are usually designed with scalability in mind. Also, the scalability of ML algorithms is not neglected. A few works (e.g. [129], [131]) study how scalable ML techniques are compared to existing methods. In turn, the implementation of novel security practices focuses on increasing network scalability as BC architectures (e.g. [150], [151]) and on how faster these methods can handle security breaches than traditional approaches (e.g. [153], [155]). Scalability requirements have also been more noticeable in Smart Cities (e.g. [165], [167]) and IIoT [185]. A fully scalable system is needed to fully provide low latency service delivery. All domains will play their part in delivering an architecture capable of handling high-demand patterns with efficient management capabilities that enable scalable deployments of future applications.

*Energy Efficiency* (C3) has been addressed in several domains. Trends on architectural paradigms such as MEC, FC and Hardware acceleration have covered energy efficiency by reducing power consumption in the infrastructure [39], network links [49] and the high performance of NNs [56], [57]. Energy-efficient hardware is a future research direction. Energy savings have also been addressed in SFC placement [76], [77] and SR performance [85]. Efficient service placement and traffic routing are also future research directions, in which service providers will focus on reducing their deployment costs while providing high QoS. The development of container-based service chains and VNFs will focus even more on minimizing resources. Works on resource allocation (e.g. [104], [105]) and caching (e.g. [121], [122]) have also studied energy consumption to deliver more efficient and greener architectures. Distributed and energy-efficient architectures are another future direction. ML has also been applied to reduce energy consumption [128], [136]. All connected devices will generate a massive volume of data that, if not properly handled, leads to slow decision-making and increased power consumption. Distributed ML operations offer an adequate solution to handle massive amounts of data. In turn, BC architectures aim to enhance performance and avoid energy waste [146], [151]. Another direction is to further optimize the performance of low-powered devices. Regarding applications, current literature focuses mostly on Smart Cities (e.g. [169], [170]) and IIoT ( [181], [182]) scenarios.

*Isolation* (C4) and NS concepts have also been getting significant attention in the last few years. However, standard implementations or detailed interfaces are still missing. Most mobile operators have already implemented PoCs to provide service isolation inside their network, but no interfaces or management models have yet been standardized. The proper administration of slice-based VNFs is still being defined by ETSI NFV MANO, including operations for the life-cycle control of slice instances. A few works address the architectural challenges of implementing isolated running environments [38] or adding NS features to their schemes [44], [48]. Also,

NS has been studied mostly for 5G (e.g. [69], [71]), but a mature design and concrete implementation guidelines are missing. Isolation has also been investigated in SFC [75], [79] and monitoring [112], [116] contexts, where increased performance has gained notable attention. Isolation offers high levels of security and TC research has been adopting isolation features to enhance current hardware platforms (e.g. [158], [159]). Emerging use cases demand QoS-specific network slices on a per-application basis, in which providers rely on fully cloud-native infrastructures to provide service isolation and meet QoS levels. Also, NS in a federation model is still unexplored. Federated slices raise further issues since multi-domain resource discovery and slice brokering are needed. How to effectively protect and secure network slices in a federation is a future direction.

*Security and Privacy* (C5) concerns are still commonly left out in most works unless security is the main focus as in BC, CS, or TC research domains. Other research fields are also tackling security issues, though no generic approach has been given. Works on architecture mostly address security risks in service offloading [38], unauthorized access [44] and secure communications [46]. Also, privacy is even more important in future networks due to the massive gathering of user data. Taleb, T. et al. [71] have addressed privacy concerns while customizing services based on NS. Their work is an adequate solution for user privacy preservation in future networks. Protect infrastructures in a continuum of virtual resources is the main research direction. BC (e.g. [149], [150], [151]) has positioned itself as the main enabler of security and privacy while providing resiliency and reliability guarantees. Combining BC concepts with existing approaches within architecture, network, and orchestration is another future direction. Lastly, research on emerging applications has studied how to enable security features without compromising performance, such as in self-driving cars [173] and IIoT (e.g. [180], [182]). Without efficient security mechanisms, users do not benefit from these use cases since safety is their main priority.

*Resilience* (C6) has been viewed as one of the main features of future architectures, but only a few works address resilience concerns. Sharma, P. K., et al. [47] have presented a distributed cloud architecture providing service guarantees when failures occur in the infrastructure. Desmouceaux, Y., et al. [88] have proposed an SR-based load balancer and its resiliency has been evaluated. Aubry, F., et al. [90] have introduced the concept of RDP, including a fault tolerance assessment. Works leveraging BC (e.g. [147], [148]) have also focused on resilience aspects. SR and BC are promising domains to efficiently provide resilience features in a cloud-native infrastructure, ensuring service continuity under critical situations. The integration of SR and BC is a future research direction. Resilience has also been studied for XR services towards avoiding playout freezes in VR adaptive streaming [178]. Emerging applications require extremely low latency but also high resilience guarantees against attacks and unexpected failures.

*Reliability* (C7) has been viewed as an important enabler of the next evolution of high-precision services. Research on novel architectures (e.g. [40], [47]) has studied mechanisms to improve service reliability. However, only simulations or small-scale scenarios have been evaluated and implementations in practical environments are missing. The reliability assessment of these approaches in large-scale scenarios is a future direction. Literature has also addressed reliability for URLLC services based on NS (e.g. [70], [72]). Also, works have investigated how to provide reliability guarantees for SFC deployments (e.g. [74], [78]) and improve network reliability focused on SR [88], [90]. SFC and SR are both future directions to improve the reliability of next-generation networks. Resource allocation research has also studied reliability improvements for D2D [100] and M2M communications [106]. In addition, autonomous cars [171], [172] and UAVs are pushing the limits of infrastructures to offer at least seven nines of reliability while precision demanding services, such as XR [176] and IIoT (e.g. [180], [181]) are stretching current infrastructures to the edge to provide high levels of reliability and large throughput data rates.

*Heterogeneity* (C8) has been addressed by several works since the heterogeneous nature of future networks is widely accepted. Architectures discuss heterogeneity based on several aspects, such as integrated network schemes [40], IoT support [43] and different computing capacities [47]. These works aim to support a large number of devices while improving network performance. Hardware-based platforms [55], [56] have also addressed heterogeneity concerns. A future direction is studying the implications of the adoption of these novel architectural paradigms. NS (e.g. [70], [72]), SFC (e.g. [75], [78]) and IBN (e.g. [81], [83]) have also addressed the interoperability between different domains and services. Novel orchestration practices (e.g. [102], [103], [114]) have also considered the heterogeneous nature of services and environments. Caching research (e.g. [119], [120]) has also addressed heterogeneity concerning distinct device (e.g. vehicles, mobile devices, BSs). DL (e.g. [126], [128]) and RL (e.g. [131], [134]) also consider different networks and architectures. In terms of applications, Smart Cities (e.g. [166], [167]) and IIoT (e.g. [180], [183]) distinct themselves due to their highly complex environment encompassing several stringent requirements. The analysis of heterogeneity implications in these scenarios is a future research direction.

*Throughput* (C9) is essential for the new range of use cases that evolve beyond 5G. MEC (e.g. [38], [37]) and FC (e.g. [47], [49]) have tackled architectural challenges to attain higher throughput under different scenarios. Hardware-accelerated platforms aim to solve performance degradation while maintaining high throughput levels (e.g. [57], [58]). NS also considers throughput as an application-specific requirement to constitute different slices [69]. SFC research (e.g. [75], [79]) has also addressed throughput under distinct chain scenarios. The impact of throughput performance should be acknowledged in future research. Resource allocation (e.g. [106], [105]) and monitoring operations (e.g. [113], [114]) have become promising areas to maximize throughput. Efficient allocation attains higher throughput levels while proper motoring identifies problems faster. ML is also providing automated allocation and monitoring operations to enhance system performance including throughput (e.g. [126], [132]). Security research (e.g. [149], [151]) has also studied how to maintain

notable throughput performance while guaranteeing system security. Regarding applications, throughput requirements are more prominent in XR since it requires real-time operations in a fully immersive environment and data cannot be compressed. Throughput requirements are expected to surpass 1 Tbps due to its data-hungry nature. Current literature (e.g. [177], [178]) has proposed efficient adaptive streaming mechanisms for VR content to improve throughput performance.

*Federation* (C10) is still unaddressed in most works. A potential research direction is the design of federated models for cloud-native infrastructures focused on the interoperability between several providers. MEC and FC have addressed architectural challenges based on multiple providers [42], [46]. Efficient SFC placement in multi-cloud environments has also been studied in [77]. IBN [80], [81] and SR [86] have also considered multi-domain environments to solve federation challenges. The design of efficient networking schemes for federated environments is a future research direction. Auto-scaling and scheduling mechanisms for cloud systems across geographically spread regions have also been studied in [111]. Developing efficient orchestration mechanisms for deploying and maintaining application components across federated clouds is another research direction. Security research has also addressed federation concepts in BC architectures [148] or multi-domain networks [156]. Existing works acknowledge the need for federation and already mention the intention to address multi-domain environments or federation concepts as future work. These extensions will lead to future federation research.

## X. Lessons Learned and Prospects

### A. Lessons Learned

Several lessons have been derived from the literature review relevant to low latency service delivery. *Architecture* has shown that MEC [37-44] and FC [45-50] are essential for the evolution of the mobile network to create a cloud-native environment where services can be deployed in a continuum of virtual resources. The integration of MEC and FiWi access networks has been studied in [40], while FC has been recently extended through connected vehicles [45]. Both works bring resources even further to the edge. Micro-services [51-54] have also revolutionized service deployments and conventional resource-hungry monoliths have been transformed into small loosely coupled micro-services. Micro-service research mostly focuses on isolation, security, and scalability aspects. Hardware acceleration [55-58] is a promising domain to mitigate performance degradation and latency introduced by network softwarization. Both [56] and [57] are adequate approaches for performance improvement of NNs. Migrating conventional hardware functions towards softwarized VNFs is needed to fully achieve low E2E latency. Hardware-accelerated NFs further help supporting low latency service delivery.

*Communication Networks* have proved that novel networking concepts provide higher levels of flexibility and scalability towards low latency service delivery. NS [69-72] addresses isolation through the creation of E2E logical slices for each group of users accessing the same service. NS has even been extended in [69] by integrating vehicles in the infrastructure. SFC [73-79] has been explored to provide a complete E2E service chain. SFC allocation has already been addressed for MEC [128] or FC [129] architectures. Recently, IBN [80-84] has been conceptualized to communicate intents to the network. Enforcing rules without detailing the system how it should perform is the main purpose of IBN research. It is still in the early stages, but IBN is an adequate answer to improve network reliability and scalability. An E2E service orchestration approach across multiple domains has been presented in [80]. The work discusses low latency challenges but also reflects on reliability and scalability issues. SR [85-90] deals with scalability concerns but also tackles resilience and reliability. Resilience has been addressed in [88]. Results have shown that IBN improves the scalability and throughput performance at a negligible cost of CPU overhead. These four networking concepts are viable alternatives to enable the support of low latency service delivery in future networks. The combination of these concepts is key to obtain a more integrated solution.

*Orchestration* practices for distributed cloud infrastructures are also gaining significant attention. Resource allocation [100-107] addresses proper service deployment, while auto-scaling [108-111] mechanisms guarantee that deployed services are sufficient to handle current network demand. Efficient provisioning strategies have been studied for vehicular networks and IoT contexts. Both [101] and [102] are adequate strategies for massive numbers of connected devices while optimizing resources in fog-cloud infrastructures. These provisioning strategies need to be complemented with auto-scaling features, such as [110] and [111], where appropriate actions are made based on the current demand. Efficient allocation and auto-scaling are only possible if proper monitoring [112-116] tools are employed. Both [113] and [115] are viable alternatives to monitor distributed systems and raise alerts in case of anomalous behaviors. Caching [117-122] also improves content delivery and reduces data transmissions if adopted at the fog or edge. Mobility and security benefit from novel caching strategies as shown in [117] and [121], respectively. All these domains will play their role in next-generation networks. However, no standardization has yet been accepted in these domains. Current works continue investigating proper solutions for efficient and cost-aware orchestration in future networks.

*ML and AI* have positioned themselves as crucial enablers of autonomous networks. DL [126-130] has already been applied to traffic congestion [126] and SFC placement and routing [127], [128]. DL has achieved high performance compared to existing methods. Recently, RL [131-136] has been applied to resource allocation and traffic control [132], [133] and auto-scaling [131], [110]. RL has shown its potential applicability in these domains due to its performance and scalability. Existing methods (e.g. ILP formulations) cannot deal with dynamic demands and their implementation in practice is difficult due to their high-resolution time. Developing RL systems capable of reallocating services in the infrastructure by reacting to sudden network demands is a major research topic in network management. Also, FL [137-141] provides

decentralized learning features for wireless networks [137], [138] alongside adequate privacy guarantees [139], [140]. All these domains will help achieve higher levels of independence in next-generation networks. The combination of these trends can lead to fully automated networks with minimum human intervention, providing self-configuration and self-repairing features that will strongly impact the performance of low latency services.

*Security and Privacy* are gaining even more importance in future networks. Traditional practices are no longer adequate since data and services are spread around in the network and stored at different levels (i.e. edge, fog, cloud). BC [146-151] provides a fully decentralized architecture capable of providing higher levels of reliability and resiliency. BC architectures as [150] and [151] have shown promising performance while mitigating security vulnerabilities. BC is one of the most promising technologies in the coming years. Novel CS [152-156] practices have also been introduced to protect infrastructures [150], mitigate attacks [152] and guarantee user privacy [156]. These works are appropriate answers to attacks or threats in the coming years. Also, TC [157-160] has been studying hardware enhancements focused on security issues. TC addresses trust while increasing hardware performance. All domains will play their role in securing distributed cloud systems enabling the support of low latency services in next-generation networks.

*Emerging Applications* are pushing towards more efficient and reliable infrastructures. Smart cities [164-170] focus mostly on stringent requirements coming from IoT. Both [165] and [167] are adequate alternatives to handle large volumes of data in a connected city. Self-driving cars [171-174] have addressed security and reliability concerns. DL has been adopted in [171] for steering prediction in autonomous cars while evasion attacks have been studied in [173]. Reliability and security are essential for the support of self-driving cars. Both works have shown promising results. XR [175-178] focuses on providing fully immersive experiences through AR glasses. XR deployments will revolutionize multimedia service delivery in future networks. MEC-based systems have already been proposed for VR content delivery [175] while a few works have studied the optimization of VR streaming systems [177], [178]. The adoption of novel architectural concepts in combination with novel streaming mechanisms is the major research direction. IIoT [179-185] has studied the adoption of IoT in the industrial sector. Low latency communications [182] and security [184] are major concerns. Reliability is also important, especially for URLLC services [180]. New ambitious and challenging use cases will emerge in the coming years leading networking innovation and the creation of new business models in next-generation networks. Prospects of emerging use cases are discussed next.

### B. Prospects of emerging use cases

All research domains introduced in this article will play a major role in enabling emerging use cases. *Architectural* paradigms such as MEC and FC will reduce latency in the communication between devices and services. Smart City and IIoT services will benefit the most from these two concepts since services are deployed at the edge and fog providing lower latency and enabling local operations. Also, micro-services allow flexible low-cost deployments as opposed to rigid costly VM allocations. Hardware-based platforms will also support efficient softwarized VNFs aiming to mitigate performance degradation.

*Communication Networks* will also play their role. Emerging use cases have diverse service requirements. XR requires throughput levels above 1Tbps, while their interactive experiences need sub-millisecond latency. In contrast, autonomous cars and UAVs demand seven levels of reliability without necessarily requiring higher throughput. NS allows setting up specific network slices for each of the envisioned use cases. Also, SFC allows developers to create service chains of containerized services for each of their applications. Container-based service chains for Smart City use cases have already been proposed in [170]. In turn, IBN and SR will completely revolutionize current networks. IBN enables network management through intents while SR provides scalable and flexible routing mechanisms, simplifying traffic engineering. Both technologies will help achieve the referenced sub-millisecond latency.

*Orchestration* practices will adapt. A plethora of allocation and auto-scaling mechanisms have been developed to address the stringent requirements of emerging applications. A trade-off between requirements is crucial for proper service allocation and scaling. XR requires low latency and high throughput, while IIoT needs high reliability and high resiliency guarantees. Cost-efficient allocations have been addressed in [101] while low latency provisioning has been studied in [102]. Anomaly detection and monitoring operations will be executed close to end-users, allowing faster responses. These methods will maintain agreed QoS levels and perform operational adjustments when needed in near real-time. Also, edge caching schemes will overcome high mobility patterns, especially necessary for self-driving cars. Cars could then access services while moving as shown in [174].

*ML and AI* will automate several tasks currently being solved via human intervention. Resource allocation [128], [130], NS [136] and privacy preservation [140] are among the most envisioned operations being resolved by these algorithms. Also, FL will contribute to an enriched collaborative learning experience where devices train a common model. This is particularly relevant for Smart City and IIoT scenarios, where a common model can increase performance and reliability.

*Security and Privacy* mechanisms will mitigate attacks and avoid service disruptions. Security is crucial for self-driving cars [173] and IIoT [184]. A failure in these scenarios may have costly repercussions (e.g. damaged machines, car accidents). TC will also keep hardware secured. Isolation guarantees will ensure secure packet processing [159]. The close interplay between all domains is key to fully support low latency 6G services in the future.

## XI. Conclusion

This article surveys the literature on ongoing research aiming to support low latency services in next-generation

networks. A taxonomy on low latency service delivery has been proposed alongside a specific set of criteria to classify research across different domains. Open challenges and future directions have been discussed, while lessons learned have been derived from our literature review. Also, prospects have been provided with a focus on the role that novel trends will play in emerging use cases such as XR.

## ACKNOWLEDGMENT

This research was performed within the project "Intelligent DEnse And Longe range IoT networks (IDEAL-IoT)" under Grant Agreement #S004017N, from the fund for Scientific Research-Flanders (FWO-V). The authors thank the FlexNGIA consortium members.

## REFERENCES

[1] D. Kreutz, F. M. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2014.

[2] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," *IEEE Communications surveys & tutorials*, vol. 18, no. 1, pp. 236–262, 2015.

[3] (2020) White paper on 6g networking. [Online]. Available: https://www.6gchannel.com/portfolio-posts/6g-white-paper-networking/

[4] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6g networks: Use cases and technologies," *IEEE Communications Magazine*, vol. 58, no. 3, pp. 55–61, 2020.

[5] N. Tijtgat, W. Van Ranst, T. Goedeme, B. Volckaert, and F. De Turck, "Embedded real-time object detection for a uav warning system," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 2110–2118.

[6] M. Ersue, "Etsi nfv management and orchestration-an overview," *Presentation at the IETF*, vol. 88, 2013.

[7] H. K. Ravuri, M. T. Vega, T. Wauters, B. Da, A. Clemm, and F. De Turck, "An experimental evaluation of flow setup latency in distributed software defined networks," in *2019 IEEE Conference on Network Softwarization (NetSoft)*. IEEE, 2019, pp. 432–437.

[8] S. R. Chowdhury, M. A. Salahuddin, N. Limam, and R. Boutaba, "Re-architecting nfv ecosystem with microservices: State of the art and research challenges," *IEEE Network*, vol. 33, no. 3, pp. 168–176, 2019.

[9] A. Gupta and R. K. Jha, "A survey of 5g network: Architecture and emerging technologies," *IEEE access*, vol. 3, pp. 1206–1232, 2015.

[10] F. Z. Yousaf, M. Bredel, S. Schaller, and F. Schneider, "Nfv and sdn—key technology enablers for 5g networks," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2468–2478, 2017.

[11] B. Blanco, J. O. Fajardo, I. Giannoulakis, E. Kafetzakis, S. Peng, J. Pérez-Romero, I. Trajkovska, P. S. Khodashenas, L. Goratti, M. Paolino *et al.*, "Technology pillars in the architecture of future 5g mobile networks: Nfv, mec and sdn," *Computer Standards & Interfaces*, vol. 54, pp. 216–228, 2017.

[12] S. Kekki, W. Featherstone, Y. Fang, P. Kuure, A. Li, A. Ranjan, D. Purkayastha, F. Jiangping, D. Frydman, G. Verin *et al.*, "Mec in 5g networks," *ETSI white paper*, vol. 28, pp. 1–28, 2018.

[13] S. Wang, J. Xu, N. Zhang, and Y. Liu, "A survey on service migration in mobile edge computing," *IEEE Access*, vol. 6, pp. 23 511–23 528, 2018.

[14] I. Parvez, A. Rahmati, I. Guvenc, A. I. Sarwat, and H. Dai, "A survey on low latency towards 5g: Ran, core network and caching solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3098–3130, 2018.

[15] T. O. Olwal, K. Djouani, and A. M. Kurien, "A survey of resource management toward 5g radio access networks," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1656–1686, 2016.

[16] F. A. Salaht, F. Desprez, and A. Lebre, "An overview of service placement problem in fog and edge computing," *ACM Computing Surveys (CSUR)*, vol. 53, no. 3, pp. 1–35, 2020.

[17] R. Mahmud, K. Ramamohanarao, and R. Buyya, "Application management in fog computing environments: A taxonomy, review and future directions," *ACM Computing Surveys (CSUR)*, vol. 53, no. 4, pp. 1–35, 2020.

[18] J. C. Guevara, R. d. S. Torres, and N. L. da Fonseca, "On the classification of fog computing applications: A machine learning perspective," *Journal of Network and Computer Applications*, vol. 159, p. 102596, 2020.

[19] P. Shantharama, A. S. Thyagaturu, and M. Reisslein, "Hardware-accelerated platforms and infrastructures for network functions: A survey of enabling technologies and research studies," *IEEE Access*, vol. 8, pp. 132 021–132 085, 2020.

[20] T. Huang, W. Yang, J. Wu, J. Ma, X. Zhang, and D. Zhang, "A survey on green 6g network: Architecture and technologies," *IEEE Access*, vol. 7, pp. 175 758–175 768, 2019.

[21] M. F. Zhani and H. ElBakoury, "Flexngia: A flexible internet architecture for the next-generation tactile internet," *Journal of Network and Systems Management*, pp. 1–45, 2020.

[22] Z. Zhang, Y. Xiao, Z. Ma, M. Xiao, Z. Ding, X. Lei, G. K. Karagiannidis, and P. Fan, "6g wireless networks: Vision, requirements, architecture, and key technologies," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 28–41, 2019.

[23] G. Javadzadeh and A. M. Rahmani, "Fog computing applications in smart cities: A systematic survey," *Wireless Networks*, vol. 26, no. 2, pp. 1433–1457, 2020.

[24] R. Hussain and S. Zeadally, "Autonomous cars: Research results, issues, and future challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1275–1313, 2018.

[25] D. Chatzopoulos, C. Bermejo, Z. Huang, and P. Hui, "Mobile augmented reality survey: From where we are to where we go," *Ieee Access*, vol. 5, pp. 6917–6950, 2017.

[26] L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on industrial informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.

[27] B. Briscoe, A. Brunstrom, A. Petlund, D. Hayes, D. Ros, J. Tsang, S. Gjessing, G. Fairhurst, C. Griwodz, and M. Welzl, "Reducing internet latency: A survey of techniques and their merits," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 2149–2196, 2014.

[28] X. Jiang, H. Shokri-Ghadikolaei, G. Fodor, E. Modiano, Z. Pang, M. Zorzi, and C. Fischione, "Low-latency networking: Where latency lurks and how to tame it," *Proceedings of the IEEE*, vol. 107, no. 2, pp. 280–306, 2018.

[29] Y.-Y. Shih, W.-H. Chung, A.-C. Pang, T.-C. Chiu, and H.-Y. Wei, "Enabling low-latency applications in fog-radio access networks," *IEEE network*, vol. 31, no. 1, pp. 52–58, 2016.

[30] J. Sachs, L. A. Andersson, J. Araújo, C. Curescu, J. Lundsjö, G. Rune, E. Steinbach, and G. Wikström, "Adaptive 5g low-latency communication for tactile internet services," *Proceedings of the IEEE*, vol. 107, no. 2, pp. 325–349, 2018.

[31] P. Marsch, I. Da Silva, O. Bulakci, M. Tesanovic, S. E. El Ayoubi, T. Rosowski, A. Kaloxylos, and M. Boldi, "5g radio access network architecture: Design guidelines and key considerations," *IEEE Communications Magazine*, vol. 54, no. 11, pp. 24–32, 2016.

[32] M. A. Habibi, M. Nasimi, B. Han, and H. D. Schotten, "A comprehensive survey of ran architectures toward 5g mobile communication system," *IEEE Access*, vol. 7, pp. 70 371–70 421, 2019.

[33] A. A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines, "5g network slicing using sdn and nfv: A survey of taxonomy, architectures and future challenges," *Computer Networks*, vol. 167, p. 106984, 2020.

[34] T. Verbelen, P. Simoens, F. De Turck, and B. Dhoedt, "Cloudlets: Bringing the cloud to the mobile user," in *Proceedings of the third ACM workshop on Mobile cloud computing and services*, 2012, pp. 29–36.

[35] J. Elmirghani, T. Klein, K. Hinton, L. Nonde, A. Lawey, T. El-Gorashi, M. Musa, and X. Dong, "Greentouch greenmeter core network energy-efficiency improvement measures and optimization," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, no. 2, pp. A250–A269, 2018.

[36] J. C. Whitson and J. E. Ramirez-Marquez, "Resiliency as a component importance measure in network reliability," *Reliability Engineering & System Safety*, vol. 94, no. 10, pp. 1685–1693, 2009.

[37] X. Liu, J. Zhang, X. Zhang, and W. Wang, "Mobility-aware coded probabilistic caching scheme for mec-enabled small cell networks," *IEEE Access*, vol. 5, pp. 17 824–17 833, 2017.

[38] L. Ma, S. Yi, N. Carter, and Q. Li, "Efficient live migration of edge services leveraging container layered storage," *IEEE Transactions on Mobile Computing*, vol. 18, no. 9, pp. 2020–2033, 2018.

[39] H.-C. Hsieh, J.-L. Chen, and A. Benslimane, "5g virtualized multi-access edge computing platform for iot applications," *Journal of Network and Computer Applications*, vol. 115, pp. 94–102, 2018.

[40] J. Liu, G. Shou, Y. Liu, Y. Hu, and Z. Guo, "Performance evaluation of integrated multi-access edge computing and fiber-wireless access networks," *IEEE Access*, vol. 6, pp. 30 269–30 279, 2018.

[41] S.-R. Yang, Y.-J. Tseng, C.-C. Huang, and W.-C. Lin, "Multi-access edge computing enhanced video streaming: Proof-of-concept implementation and prediction/qoe models," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1888–1902, 2018.

[42] S. D. A. Shah, M. A. Gregory, S. Li, and R. D. R. Fontes, "Sdn enhanced multi-access edge computing (mec) for e2e mobility and qos management," *IEEE Access*, vol. 8, pp. 77 459–77 469, 2020.

[43] P. Ranaweera, V. N. Imrith, M. Liyanag, and A. D. Jurcut, "Security as a service platform leveraging multi-access edge computing infrastructure provisions," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–6.

[44] A. Ksentini and P. A. Frangoudis, "Toward slicing-enabled multi-access edge computing in 5g," *IEEE Network*, vol. 34, no. 2, pp. 99–105, 2020.

[45] M. Sookhak, F. R. Yu, Y. He, H. Talebian, N. S. Safa, N. Zhao, M. K. Khan, and N. Kumar, "Fog vehicular computing: Augmentation of fog computing using vehicular cloud computing," *IEEE Vehicular Technology Magazine*, vol. 12, no. 3, pp. 55–64, 2017.

[46] R. Moreno-Vozmediano, R. S. Montero, E. Huedo, and I. M. Llorente, "Cross-site virtual network in cloud and fog computing," *IEEE Cloud Computing*, vol. 4, no. 2, pp. 46–53, 2017.

[47] P. K. Sharma, M.-Y. Chen, and J. H. Park, "A software defined fog node based distributed blockchain cloud architecture for iot," *Ieee Access*, vol. 6, pp. 115–124, 2017.

[48] R. Bruschi, F. Davoli, P. Lago, and J. F. Pajo, "A scalable sdn slicing scheme for multi-domain fog/cloud services," in *2017 IEEE Conference on Network Softwarization (NetSoft)*. IEEE, 2017, pp. 1–6.

[49] E. Baccarelli, P. G. V. Naranjo, M. Scarpiniti, M. Shojafar, and J. H. Abawajy, "Fog of everything: Energy-efficient networked computing architectures, research challenges, and a case study," *IEEE access*, vol. 5, pp. 9882–9910, 2017.

[50] J. Santos, T. Wauters, B. Volckaert, and F. De Turck, "Fog computing: Enabling the management and orchestration of smart city applications in 5g networks," *Entropy*, vol. 20, no. 1, p. 4, 2018.

[51] S. Brenner, T. Hundt, G. Mazzeo, and R. Kapitza, "Secure cloud micro services using intel sgx," in *IFIP International Conference on Distributed Applications and Interoperable Systems*. Springer, 2017, pp. 177–191.

[52] D. Guija and M. S. Siddiqui, "Identity and access control for micro-services based 5g nfv platforms," in *Proceedings of the 13th International Conference on Availability, Reliability and Security*, 2018, pp. 1–10.

[53] R. Xu, S. Y. Nikouei, Y. Chen, E. Blasch, and A. Aved, "Blendmas: A blockchain-enabled decentralized microservices architecture for smart public safety," in *2019 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 2019, pp. 564–571.

[54] O. Debauche, S. Mahmoudi, S. A. Mahmoudi, P. Manneback, and F. Lebeau, "A new edge architecture for ai-iot services deployment," *Procedia Computer Science*, vol. 175, pp. 10–19, 2020.

[55] X. Zhang, X. Shao, G. Provelengios, N. K. Dumpala, L. Gao, and R. Tessier, "Scalable network function virtualization for heterogeneous middleboxes," in *2017 IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*. IEEE, 2017, pp. 219–226.

[56] Y. Umuroglu, N. J. Fraser, G. Gambardella, M. Blott, P. Leong, M. Jahre, and K. Vissers, "Finn: A framework for fast, scalable binarized neural network inference," in *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, 2017, pp. 65–74.

[57] R. Cai, A. Ren, N. Liu, C. Ding, L. Wang, X. Qian, M. Pedram, and Y. Wang, "Vibnn: Hardware acceleration of bayesian neural networks," *ACM SIGPLAN Notices*, vol. 53, no. 2, pp. 476–488, 2018.

[58] M. Owaida, G. Alonso, L. Fogliarini, A. Hock-Koon, and P.-E. Melet, "Lowering the latency of data processing pipelines through fpga based hardware acceleration," *Proceedings of the VLDB Endowment*, vol. 13, no. 1, pp. 71–85, 2019.

[59] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5g network edge cloud architecture and orchestration," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.

[60] (2016) Etsi technical specification, mobile edge computing (mec): Framework and reference architecture. [Online]. Available: http://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf

[61] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, "A comprehensive survey on fog computing: State-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 416–464, 2017.

[62] P. Bellavista, J. Berrocal, A. Corradi, S. K. Das, L. Foschini, and A. Zanni, "A survey on fog computing for the internet of things," *Pervasive and mobile computing*, vol. 52, pp. 71–99, 2019.

[63] N. Parvez, A. Mahanti, and C. Williamson, "An analytic throughput model for tcp newreno," *IEEE/ACM Transactions on Networking*, vol. 18, no. 2, pp. 448–461, 2009.

[64] (2016) Etsi technical specification, onem2m functional architecture, onem2m ts-0001 version 2.10.0 release 2. [Online]. Available: http://www.etsi.org/deliver/etsi_ts/118100_118199/118101/02.10.00_60/ts_118101v021000p.pdf

[65] I. Nadareishvili, R. Mitra, M. McLarty, and M. Amundsen, *Microservice architecture: aligning principles, practices, and culture*. " O'Reilly Media, Inc.", 2016.

[66] M. A. Christie, A. Bhandar, S. Nakandala, S. Marru, E. Abeysinghe, S. Pamidighantam, and M. E. Pierce, "Using keycloak for gateway authentication and authorization," 2017.

[67] B. Burns, B. Grant, D. Oppenheimer, E. Brewer, and J. Wilkes, "Borg, omega, and kubernetes," *Queue*, vol. 14, no. 1, pp. 70–93, 2016.

[68] L. Linguaglossa, S. Lange, S. Pontarelli, G. Rétvári, D. Rossi, T. Zinner, R. Bifulco, M. Jarschel, and G. Bianchi, "Survey of performance acceleration techniques for network function virtualization," *Proceedings of the IEEE*, vol. 107, no. 4, pp. 746–764, 2019.

[69] C. Campolo, A. Molinaro, A. Iera, and F. Menichella, "5g network slicing for vehicle-to-everything services," *IEEE Wireless Communications*, vol. 24, no. 6, pp. 38–45, 2017.

[70] A. Ksentini and N. Nikaein, "Toward enforcing network slicing on ran: Flexibility and resources abstraction," *IEEE Communications Magazine*, vol. 55, no. 6, pp. 102–108, 2017.

[71] T. Taleb, B. Mada, M.-I. Corici, A. Nakao, and H. Flinck, "Permit: Network slicing for personalized 5g mobile telecommunications," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 88–93, 2017.

[72] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5g wireless network slicing for embb, urllc, and mmtc: A communication-theoretic view," *Ieee Access*, vol. 6, pp. 55 765–55 779, 2018.

[73] L. Qu, C. Assi, K. Shaban, and M. J. Khabbaz, "A reliability-aware network service chain provisioning with delay guarantees in nfv-enabled enterprise datacenter networks," *IEEE Transactions on Network and Service Management*, vol. 14, no. 3, pp. 554–568, 2017.

[74] L. Zhang, S. Lai, C. Wu, Z. Li, and C. Guo, "Virtualized network coding functions on the internet," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2017, pp. 129–139.

[75] I. Trajkovska, M.-A. Kourtis, C. Sakkas, D. Baudinot, J. Silva, P. Harsh, G. Xylouris, T. M. Bohnert, and H. Koumaras, "Sdn-based service function chaining mechanism and service prototype implementation in nfv scenario," *Computer Standards & Interfaces*, vol. 54, pp. 247–265, 2017.

[76] I. Jang, D. Suh, S. Pack, and G. Dán, "Joint optimization of service function placement and flow distribution for service function chaining," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2532–2541, 2017.

[77] D. Bhamare, M. Samaka, A. Erbad, R. Jain, L. Gupta, and H. A. Chan, "Optimal virtual network function placement in multi-cloud service function chaining architecture," *Computer Communications*, vol. 102, pp. 1–16, 2017.

[78] H. Hawilo, M. Jammal, and A. Shami, "Network function virtualization-aware orchestrator for service function chaining placement in the cloud," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 3, pp. 643–655, 2019.

[79] Z. Xiang, F. Gabriel, E. Urbano, G. T. Nguyen, M. Reisslein, and F. H. Fitzek, "Reducing latency in virtual machines: Enabling tactile internet for human-machine co-working," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 5, pp. 1098–1116, 2019.

[80] W. Cerroni, C. Buratti, S. Cerboni, G. Davoli, C. Contoli, F. Foresta, F. Callegati, and R. Verdone, "Intent-based management and orchestration of heterogeneous openflow/iot sdn domains," in *2017 IEEE*

*Conference on Network Softwarization (NetSoft)*. IEEE, 2017, pp. 1–9.

[81] S. Arezoumand, K. Dzeparoska, H. Bannazadeh, and A. Leon-Garcia, "Md-idn: Multi-domain intent-driven networking in software-defined infrastructures," in *2017 13th International Conference on Network and Service Management (CNSM)*. IEEE, 2017, pp. 1–7.

[82] T. Szyrkowiec, M. Santuari, M. Chamania, D. Siracusa, A. Autenrieth, V. Lopez, J. Cho, and W. Kellerer, "Automatic intent-based secure service creation through a multilayer sdn network orchestration," *Journal of Optical Communications and Networking*, vol. 10, no. 4, pp. 289–297, 2018.

[83] K. Abbas, M. Afaq, T. Ahmed Khan, A. Rafiq, and W.-C. Song, "Slicing the core network and radio access network domains through intent-based networking for 5g networks," *Electronics*, vol. 9, no. 10, p. 1710, 2020.

[84] Y. Wang, Z. Tian, Y. Sun, X. Du, and N. Guizani, "Locjury: An ibn-based location privacy preserving scheme for iocv," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[85] J. Pang, G. Xu, and X. Fu, "Sdn-based data center networking with collaboration of multipath tcp and segment routing," *IEEE Access*, vol. 5, pp. 9764–9773, 2017.

[86] A. Giorgetti, A. Sgambelluri, F. Paolucci, F. Cugini, and P. Castoldi, "Segment routing for effective recovery and multi-domain traffic engineering," *Journal of Optical Communications and Networking*, vol. 9, no. 2, pp. A223–A232, 2017.

[87] A. Cianfrani, M. Listanti, and M. Polverini, "Incremental deployment of segment routing into an isp network: A traffic engineering perspective," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 3146–3160, 2017.

[88] Y. Desmouceaux, P. Pfister, J. Tollet, M. Townsley, and T. Clausen, "6lb: Scalable and application-aware load balancing with segment routing," *IEEE/ACM Transactions on Networking*, vol. 26, no. 2, pp. 819–834, 2018.

[89] U. Chunduri, A. Clemm, and R. Li, "Preferred path routing-a next-generation routing framework beyond segment routing," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–7.

[90] F. Aubry, S. Vissicchio, O. Bonaventure, and Y. Deville, "Robustly disjoint paths with segment routing," in *Proceedings of the 14th international conference on emerging networking experiments and technologies*, 2018, pp. 204–216.

[91] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, "Network slicing and softwarization: A survey on principles, enabling technologies, and solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2429–2453, 2018.

[92] J. Santos, T. Wauters, B. Volckaert, and F. De Turck, "Towards delay-aware container-based service function chaining in fog computing," in *NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2020, pp. 1–9.

[93] H. Moens and F. De Turck, "Vnf-p: A model for efficient placement of virtualized network functions," in *10th International Conference on Network and Service Management (CNSM) and Workshop*. IEEE, 2014, pp. 418–423.

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

[95] E. Zeydan and Y. Turk, "Recent advances in intent-based networking: A survey," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*. IEEE, 2020, pp. 1–5.

[96] A. Clemm, L. Ciavaglia, L. Granville, and J. Tantsura. (2020) Intent-based networking-concepts and definitions. [Online]. Available: https://tools.ietf.org/pdf/draft-irtf-nmrg-ibn-concepts-definitions-02.pdf

[97] P. Berde, M. Gerola, J. Hart, Y. Higuchi, M. Kobayashi, T. Koide, B. Lantz, B. O'Connor, P. Radoslavov, W. Snow *et al.*, "Onos: towards an open, distributed sdn os," in *Proceedings of the third workshop on Hot topics in software defined networking*, 2014, pp. 1–6.

[98] Z. N. Abdullah, I. Ahmad, and I. Hussain, "Segment routing in software defined networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 464–486, 2018.

[99] M. Mitzenmacher, "The power of two choices in randomized load balancing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 12, no. 10, pp. 1094–1104, 2001.

[100] L. Liang, G. Y. Li, and W. Xu, "Resource allocation for d2d-enabled vehicular communications," *IEEE Transactions on Communications*, vol. 65, no. 7, pp. 3186–3197, 2017.

[101] H. R. Arkian, A. Diyanat, and A. Pourkhalili, "Mist: Fog-based data analytics scheme with cost-efficient resource provisioning for iot crowdsensing applications," *Journal of Network and Computer Applications*, vol. 82, pp. 152–165, 2017.

[102] J. Santos, T. Wauters, B. Volckaert, and F. De Turck, "Resource provisioning for iot application services in smart cities," in *2017 13th International Conference on Network and Service Management (CNSM)*. IEEE, 2017, pp. 1–9.

[103] J. Yao and N. Ansari, "Qos-aware fog resource provisioning and mobile device power control in iot networks," *IEEE Transactions on Network and Service Management*, vol. 16, no. 1, pp. 167–175, 2018.

[104] P. Podili and K. Kataoka, "Effective resource provisioning for qos-aware virtual networks in sdn," in *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2018, pp. 1–9.

[105] H. Zhang, F. Fang, J. Cheng, K. Long, W. Wang, and V. C. Leung, "Energy-efficient resource allocation in noma heterogeneous networks," *IEEE Wireless Communications*, vol. 25, no. 2, pp. 48–53, 2018.

[106] Z. Zhou, Y. Guo, Y. He, X. Zhao, and W. M. Bazzi, "Access control and resource allocation for m2m communications in industrial automation," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 3093–3103, 2019.

[107] A. Farhad, D.-H. Kim, B.-H. Kim, A. F. Y. Mohammed, and J.-Y. Pyun, "Mobility-aware resource assignment to iot applications in long-range wide area networks," *IEEE Access*, vol. 8, pp. 186 111–186 124, 2020.

[108] M. S. Aslanpour, M. Ghobaei-Arani, and A. N. Toosi, "Auto-scaling web applications in clouds: A cost-aware approach," *Journal of Network and Computer Applications*, vol. 95, pp. 26–41, 2017.

[109] S. Rahman, T. Ahmed, M. Huynh, M. Tornatore, and B. Mukherjee, "Auto-scaling vnfs using machine learning to improve qos and reduce cost," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–6.

[110] D. Lee, J.-H. Yoo, and J. W.-K. Hong, "Deep q-networks based auto-scaling for service function chaining," in *2020 16th International Conference on Network and Service Management (CNSM)*. IEEE, 2020, pp. 1–9.

[111] T. Lin and A. Leon-Garcia, "Towards a client-centric qos auto-scaling system," in *NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2020, pp. 1–9.

[112] F. Moradi, C. Flinta, A. Johnsson, and C. Meirosu, "Conmon: An automated container based network performance monitoring system," in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. IEEE, 2017, pp. 54–62.

[113] G. Tangari, D. Tuncer, M. Charalambides, and G. Pavlou, "Decentralized monitoring for large-scale software-defined networks," in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. IEEE, 2017, pp. 289–297.

[114] S. Y. Shah, Z. Yuan, S. Lu, and P. Zerfos, "Dependency analysis of cloud applications for performance monitoring using recurrent neural networks," in *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017, pp. 1534–1543.

[115] D. Perdices, D. Muelas, L. de Pedro, and J. E. L. de Vergara, "Network performance monitoring with flexible models of multi-point passive measurements," in *2018 14th International Conference on Network and Service Management (CNSM)*. IEEE, 2018, pp. 1–9.

[116] I. J. Sanz, D. M. F. Mattos, and O. C. M. B. Duarte, "Sfcperf: An automatic performance evaluation framework for service function chaining," in *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2018, pp. 1–9.

[117] M. Chen, Y. Hao, L. Hu, K. Huang, and V. K. Lau, "Green and mobility-aware caching in 5g networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 12, pp. 8347–8361, 2017.

[118] K. Tantayakul, R. Dhaou, and B. Paillassa, "Mobility management with caching policy over sdn architecture," in *2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. IEEE, 2017, pp. 1–7.

[119] K. Zhang, S. Leng, Y. He, S. Maharjan, and Y. Zhang, "Cooperative content caching in 5g networks with mobile edge computing," *IEEE Wireless Communications*, vol. 25, no. 3, pp. 80–87, 2018.

[120] Y. Hao, M. Chen, L. Hu, M. S. Hossain, and A. Ghoneim, "Energy efficient task caching and offloading for mobile edge computing," *IEEE Access*, vol. 6, pp. 11 365–11 373, 2018.

[121] L. Xiao, X. Wan, C. Dai, X. Du, X. Chen, and M. Guizani, "Security in mobile edge caching with reinforcement learning," *IEEE Wireless Communications*, vol. 25, no. 3, pp. 116–122, 2018.

[122] F. Cheng, G. Gui, N. Zhao, Y. Chen, J. Tang, and H. Sari, "Uav-relaying-assisted secure transmission with caching," *IEEE Transactions on Communications*, vol. 67, no. 5, pp. 3140–3153, 2019.

[123] L. Deboosere, B. Vankeirsbilck, P. Simoens, F. De Turck, B. Dhoedt, and P. Demeester, "Efficient resource management for virtual desktop cloud computing," *The Journal of Supercomputing*, vol. 62, no. 2, pp. 741–767, 2012.

[124] J. G. Herrera and J. F. Botero, "Resource allocation in nfv: A comprehensive survey," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 518–532, 2016.

[125] C. Qu, R. N. Calheiros, and R. Buyya, "Auto-scaling web applications in clouds: A taxonomy and survey," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, pp. 1–33, 2018.

[126] F. Tang, Z. M. Fadlullah, B. Mao, and N. Kato, "An intelligent traffic load prediction-based adaptive channel assignment algorithm in sdn-iot: A deep learning approach," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 5141–5154, 2018.

[127] J. Pei, P. Hong, and D. Li, "Virtual network function selection and chaining based on deep learning in sdn and nfv-enabled networks," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2018, pp. 1–6.

[128] F. Jiang, K. Wang, L. Dong, C. Pan, W. Xu, and K. Yang, "Deep learning based joint resource scheduling algorithms for hybrid mec networks," *IEEE Internet of Things Journal*, 2019.

[129] N. Siasi, M. Jasim, A. Aldalbahi, and N. Ghani, "Deep learning for service function chain provisioning in fog computing," *IEEE Access*, vol. 8, pp. 167 665–167 683, 2020.

[130] Z. Ali, S. Khaf, Z. H. Abbas, G. Abbas, F. Muhammad, and S. Kim, "A deep learning approach for mobility-aware and energy-efficient resource allocation in mec," *IEEE Access*, vol. 8, pp. 179 530–179 546, 2020.

[131] L. Chen, J. Lingys, K. Chen, and F. Liu, "Auto: Scaling deep reinforcement learning for datacenter-scale automatic traffic optimization," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, 2018, pp. 191–205.

[132] Z. Xu, J. Tang, J. Meng, W. Zhang, Y. Wang, C. H. Liu, and D. Yang, "Experience-driven networking: A deep reinforcement learning based approach," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 1871–1879.

[133] J. Li, H. Gao, T. Lv, and Y. Lu, "Deep reinforcement learning based computation offloading and resource allocation for mec," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2018, pp. 1–6.

[134] R. Li, Z. Zhao, Q. Sun, I. Chih-Lin, C. Yang, X. Chen, M. Zhao, and H. Zhang, "Deep reinforcement learning for resource management in network slicing," *IEEE Access*, vol. 6, pp. 74 429–74 441, 2018.

[135] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep reinforcement learning based resource allocation for v2v communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3163–3173, 2019.

[136] G. Faraci, C. Grasso, and G. Schembra, "Design of a 5g network slice extension with mec uavs managed with reinforcement learning," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 10, pp. 2356–2371, 2020.

[137] N. H. Tran, W. Bao, A. Zomaya, N. M. NH, and C. S. Hong, "Federated learning over wireless networks: Optimization model design and analysis," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 1387–1395.

[138] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Transactions on Wireless Communications*, 2020.

[139] Y. Qu, L. Gao, T. H. Luan, Y. Xiang, S. Yu, B. Li, and G. Zheng, "Decentralized privacy using blockchain-enabled federated learning in fog computing," *IEEE Internet of Things Journal*, 2020.

[140] C. Zhou, A. Fu, S. Yu, W. Yang, H. Wang, and Y. Zhang, "Privacy-preserving federated learning in fog computing," *IEEE Internet of Things Journal*, 2020.

[141] J. Kang, Z. Xiong, D. Niyato, Y. Zou, Y. Zhang, and M. Guizani, "Reliable federated learning for mobile networks," *IEEE Wireless Communications*, vol. 27, no. 2, pp. 72–80, 2020.

[142] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1, no. 2.

[143] J. Santos, T. Wauters, B. Volckaert, and F. De Turck, "Reinforcement learning for service function chain allocation in fog computing," *Book Chapter in revision, Submitted to Communications Network and Service Management In the Era of Artificial Intelligence and Machine Learning, IEEE Press*, 2020.

[144] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*, 2016, pp. 1928–1937.

[145] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–19, 2019.

[146] A. Dorri, S. S. Kanhere, and R. Jurdak, "Towards an optimized blockchain for iot," in *2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI)*. IEEE, 2017, pp. 173–178.

[147] M. Zamani, M. Movahedi, and M. Raykova, "Rapidchain: Scaling blockchain via full sharding," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 931–948.

[148] I. D. Alvarenga, G. A. Rebello, and O. C. M. Duarte, "Securing configuration management and migration of virtual network functions using blockchain," in *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2018, pp. 1–9.

[149] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, "Untangling blockchain: A data processing view of blockchain systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 7, pp. 1366–1385, 2018.

[150] W. Zheng, Z. Zheng, X. Chen, K. Dai, P. Li, and R. Chen, "Nutbaas: A blockchain-as-a-service platform," *Ieee Access*, vol. 7, pp. 134 422–134 433, 2019.

[151] Z. Qiu, J. Hao, Y. Guo, and Y. Zhang, "Dual vote confirmation based consensus design for blockchain integrated iot," in *NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2020, pp. 1–7.

[152] P. Varga, G. Kathareios, Á. Máté, R. Clauberg, A. Anghel, P. Orosz, B. Nagy, T. Tóthfalusi, L. Kovács, and M. Gusat, "Real-time security services for sdn-based datacenters," in *2017 13th International Conference on Network and Service Management (CNSM)*. IEEE, 2017, pp. 1–9.

[153] A. A. Diro, N. Chilamkurti, and N. Kumar, "Lightweight cybersecurity schemes using elliptic curve cryptography in publish-subscribe fog computing," *Mobile Networks and Applications*, vol. 22, no. 5, pp. 848–858, 2017.

[154] A. S. Sohal, R. Sandhu, S. K. Sood, and V. Chang, "A cybersecurity framework to identify malicious edge device in fog computing and cloud-of-things environments," *Computers & Security*, vol. 74, pp. 340–354, 2018.

[155] K. Dzeparoska, H. Bannazadeh, and A. Leon-Garcia, "Sdx-based security collaboration: Extending the security reach beyond network domains," in *2018 14th International Conference on Network and Service Management (CNSM)*. IEEE, 2018, pp. 63–71.

[156] K. D. Joshi and K. Kataoka, "Prime-q: Privacy aware end-to-end qos framework in multi-domain sdn," in *2019 IEEE Conference on Network Softwarization (NetSoft)*. IEEE, 2019, pp. 169–177.

[157] S. Aga and S. Narayanasamy, "Invisimem: Smart memory for trusted computing," in *International Symposium on Computer Architecture*, vol. 10, no. 3079856.3080232, 2017.

[158] A. T. Gjerdrum, R. Pettersen, H. D. Johansen, and D. Johansen, "Performance of trusted computing in cloud infrastructures with intel sgx." in *CLOSER*, 2017, pp. 668–675.

[159] M. Coughlin, E. Keller, and E. Wustrow, "Trusted click: Overcoming security issues of nfv in the cloud," in *Proceedings of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, 2017, pp. 31–36.

[160] H. Yin, D. Guo, K. Wang, Z. Jiang, Y. Lyu, and J. Xing, "Hyper-connected network: A decentralized trusted computing and networking paradigm," *IEEE Network*, vol. 32, no. 1, pp. 112–117, 2018.

[161] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, "Blockchain," *Business & Information Systems Engineering*, vol. 59, no. 3, pp. 183–187, 2017.

[162] R. Von Solms and J. Van Niekerk, "From information security to cyber security," *computers & security*, vol. 38, pp. 97–102, 2013.

[163] S. W. Smith, *Trusted computing platforms: design and applications*. Springer, 2013.

[164] F. Montori, L. Bedogni, and L. Bononi, "A collaborative internet of things architecture for smart cities and environmental monitoring," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 592–605, 2017.

[165] B. Cheng, G. Solmaz, F. Cirillo, E. Kovacs, K. Terasawa, and A. Kitazawa, "Fogflow: Easy programming of iot services over cloud and edges for smart cities," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 696–707, 2017.

[166] W. Ejaz, M. Naeem, A. Shahid, A. Anpalagan, and M. Jo, "Efficient energy management for the internet of things in smart cities," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 84–91, 2017.
[167] J. Santos, T. Vanhove, M. Sebrechts, T. Dupont, W. Kerckhove, B. Braem, G. Van Seghbroeck, T. Wauters, P. Leroux, S. Latre *et al.*, "City of things: Enabling resource provisioning in smart cities," *IEEE Communications Magazine*, vol. 56, no. 7, pp. 177–183, 2018.
[168] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge computing for iot-based energy management in smart cities," *IEEE Network*, vol. 33, no. 2, pp. 111–117, 2019.
[169] M. Aloqaily, S. Otoum, I. Al Ridhawi, and Y. Jararweh, "An intrusion detection system for connected vehicles in smart cities," *Ad Hoc Networks*, vol. 90, p. 101842, 2019.
[170] J. Santos, T. Wauters, B. Volckaert, and F. De Turck, "Towards end-to-end resource provisioning in fog computing over low power wide area networks," *Journal of Network and Computer Applications*, p. 102915, 2020.
[171] A. I. Maqueda, A. Loquercio, G. Gallego, N. García, and D. Scaramuzza, "Event-based vision meets deep learning on steering prediction for self-driving cars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5419–5427.
[172] S. Chen, Y. Chen, S. Zhang, and N. Zheng, "A novel integrated simulation and testing platform for self-driving cars with hardware in the loop," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 3, pp. 425–436, 2019.
[173] A. Chernikova, A. Oprea, C. Nita-Rotaru, and B. Kim, "Are self-driving cars secure? evasion attacks against deep neural networks for steering angle prediction," in *2019 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2019, pp. 132–137.
[174] A. Ndikumana, N. H. Tran, K. T. Kim, C. S. Hong *et al.*, "Deep learning based caching for self-driving cars in multi-access edge computing," *IEEE Transactions on Intelligent Transportation Systems*, 2020.
[175] Y. Liu, J. Liu, A. Argyriou, and S. Ci, "Mec-assisted panoramic vr video streaming over millimeter wave mobile networks," *IEEE Transactions on Multimedia*, vol. 21, no. 5, pp. 1302–1316, 2018.
[176] A. Doumanoglou, D. Griffin, J. Serrano, N. Zioulis, T. K. Phan, D. Jiménez, D. Zarpalas, F. Alvarez, M. Rio, and P. Daras, "Quality of experience for 3-d immersive media streaming," *IEEE Transactions on Broadcasting*, vol. 64, no. 2, pp. 379–391, 2018.
[177] J. van der Hooft, M. T. Vega, S. Petrangeli, T. Wauters, and F. De Turck, "Optimizing adaptive tile-based virtual reality video streaming," in *2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. IEEE, 2019, pp. 381–387.
[178] J. van der Hooft, T. Wauters, F. De Turck, C. Timmerer, and H. Hellwagner, "Towards 6dof http adaptive streaming through point cloud compression," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 2405–2413.
[179] L. Zhou, D. Wu, J. Chen, and Z. Dong, "When computation hugs intelligence: Content-aware data processing for industrial iot," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1657–1666, 2017.
[180] J. Cheng, W. Chen, F. Tao, and C.-L. Lin, "Industrial iot in 5g environment towards smart manufacturing," *Journal of Industrial Information Integration*, vol. 10, pp. 10–19, 2018.
[181] M. Luvisotto, F. Tramarin, L. Vangelista, and S. Vitturi, "On the use of lorawan for indoor industrial iot applications," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.
[182] J. Hiller, M. Henze, M. Serror, E. Wagner, J. N. Richter, and K. Wehrle, "Secure low latency communication for constrained industrial iot scenarios," in *2018 IEEE 43rd Conference on Local Computer Networks (LCN)*. IEEE, 2018, pp. 614–622.
[183] H. Yang, A. Alphones, W.-D. Zhong, C. Chen, and X. Xie, "Learning-based energy-efficient resource management by heterogeneous rf/vlc for ultra-reliable low-latency industrial iot networks," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 8, pp. 5565–5576, 2019.
[184] S. R. Niya, E. Schiller, I. Cepilov, and B. Stiller, "Biit: Standardization of blockchain-based i 2 ot systems in the i4 era," in *NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2020, pp. 1–9.
[185] T. Kumar, E. Harjula, M. Ejaz, A. Manzoor, P. Porambage, I. Ahmad, M. Liyanage, A. Braeken, and M. Ylianttila, "Blockedge: Blockchain-edge framework for industrial iot networks," *IEEE Access*, vol. 8, pp. 154 166–154 185, 2020.
[186] T.-h. Kim, C. Ramos, and S. Mohammed, "Smart city and iot," 2017.
[187] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. M. Paixão, F. Mutz *et al.*, "Self-driving cars: A survey," *Expert Systems with Applications*, p. 113816, 2020.
[188] Å. Fast-Berglund, L. Gong, and D. Li, "Testing and validating extended reality (xr) technologies in manufacturing," *Procedia Manufacturing*, vol. 25, pp. 31–38, 2018.
[189] C. Perera, C. H. Liu, S. Jayawardena, and M. Chen, "A survey on internet of things from industrial market perspective," *IEEE Access*, vol. 2, pp. 1660–1679, 2014.

**José Santos** obtained his M.Sc. degree in Electrical and Computers Engineering in July 2015 from the University of Porto, Portugal. He is currently a Ph.D. Student of Computer Science in the Internet Technology and Data Science Lab (IDLab) Research Group at Ghent University - imec, Belgium. Before joining IDLab, he was a Research Intern at PROEF Group where he was involved in EU-funded projects. His research interests include Cloud and Fog Computing, IoT, Software-Defined Networking, Service Function Chaining, and Reinforcement Learning.

**Tim Wauters** obtained his M.Sc. and PhD degrees in electro-technical engineering from Ghent University in 2001 and 2007 respectively. He has been working as a post-doctoral fellow of the F.W.O.-V. in the Department of Information Technology (INTEC) at Ghent University, and is now also active as a senior researcher at imec. His main research interests focus on the design and management of networked services, covering multimedia distribution, cybersecurity, big data and smart cities. His work has been published in more than 130 scientific publications.

**Bruno Volckaert** is professor advanced programming and software engineering at Ghent University and senior researcher at imec. In 2006 he obtained his PhD on resource management for Grid computing. His current research deals with reliable and high performance distributed software systems. He has worked on over 45 national and international research projects and is author or co-author of more than 145 peer-reviewed papers published in international journals and conference proceedings.

**Filip De Turck** leads the network and service management research group at Ghent University, Belgium and imec. He (co-) authored over 700 peer reviewed papers and his research interests include design of efficient softwarized network and cloud systems. He is involved in several research projects with industry and academia, serves as chair of the IEEE Technical Committee on Network Operations and Management (CNOM), and steering committee member of the IM, NOMS, CNSM and NetSoft conferences. Prof. Filip De Turck serves as Editor-in-Chief of IEEE Transactions on Network and Service Management (TNSM), and was recently named an IEEE Fellow.