WARSAW UNIVERSITY OF TECHNOLOGY

Ph.D. Thesis

Information and Communication Technology/
Engineering and Technology

Robert, Kołakowski M.Sc.

Data Plane Traffic Engineering Algorithms Supporting Next-Generation Mobile Networks

Supervisors
Prof. DSc., Eng. **Krzysztof, Szczypiorski**

Assistant Supervisor
PhD, Eng., **Sławomir, Kukliński**

Warsaw 2026

Abstract

This thesis presents the results of theoretical and practical research on various aspects related to Software-Defined Networking (SDN) and SDN Traffic Engineering (SDN-TE) in next-generation mobile networks. The evolution of mobile networks towards 6th Generation (6G) introduces new requirements that require improved networking solutions that provide flexible, intelligent, and scalable control mechanisms. Software-Defined Networking (SDN) is a promising enabler of these capabilities, thanks to its inherent programmability, which facilitates performance optimization via SDN Traffic Engineering (SDN-TE). However, the scalability limitations of conventional SDN architectures hinder their deployment in carrier-grade mobile networks.

This work addresses these challenges by investigating scalable SDN solutions for future mobile networks. It focuses on three main research directions:

- the design of scalable SDN architecture suitable for large-scale distributed networks,
- the development of scalable and intelligent SDN-TE algorithms for efficient network optimization in distributed network environments,
- the exploration of novel SDN and SDN-TE applications that demonstrate the benefits of programmable networking in the 6G era.

The presented achievements contribute to the state of knowledge in the discipline of *Information and Communication Technology*. The results span multiple aspects of communication networks, including their design, implementation, optimization, and application in the broader context of 6G networks.

Keywords: 6G, software-defined networking, traffic engineering, distributed systems, artificial intelligence, deep reinforcement learning, multi-agent systems, mobile networks, user plane, core network, management & orchestration

Streszczenie

Niniejsza rozprawa przedstawia wyniki badań teoretycznych i praktycznych dotyczących różnych aspektów programowalnych sieci komputerowych (ang. *Software-Defined Networking, SDN*) oraz inżynierii ruchu w sieciach SDN (ang. *SDN Traffic Engineering, SDN-TE*) w kontekście sieci mobilnych nowej generacji. Ewolucja sieci mobilnych w kierunku 6G wprowadza nowe wymagania, które wymagają wdrożenia zaawansowanych mechanizmów zapewniających inteligentne, elastyczne i skalowalne sterowanie ruchem w sieci. Koncepcja SDN stanowi obiecujące rozwiązanie umożliwiające realizację tych założeń dzięki programowalności, która pozwala optymalizować działanie sieci za pomocą mechanizmów SDN-TE. Jednakże oryginalna architektura SDN jest nieskalowalna, co uniemożliwia jej wdrażanie w sieciach produkcyjnych.

W ramach pracy podjęto te wyzwania poprzez badanie skalowalnych rozwiązań SDN dla przyszłych sieci mobilnych. Badania koncentrują się na trzech głównych kierunkach:

- projektowaniu skalowalnej architektury SDN odpowiedniej dla wielkoskalowych sieci rozproszonych,
- opracowaniu skalowalnych i inteligentnych algorytmów SDN-TE przeznaczonych do efektywnej optymalizacji sieci w środowiskach rozproszonych,
- badaniu nowych zastosowań SDN oraz SDN-TE, które demonstrują korzyści płynące z programowalnych sieci w erze 6G.

Przedstawione osiągnięcia przyczyniają się do rozwoju stanu wiedzy w dyscyplinie *Informatyka Techniczna i Telekomunikacja*. Uzyskane wyniki obejmują różnorodne aspekty sieci komunikacyjnych, w tym ich projektowanie, implementację, optymalizację oraz zastosowanie w szerszym kontekście sieci 6G.

Słowa kluczowe: 6G, programowalne sieci komputerowe, inżynieria ruchu, systemy rozproszone, płaszczyzna użytkownika, sieci mobilne, sieć rdzeniowa, sztuczna inteligencja, głębokie uczenie ze wzmocnieniem, systemy wieloagentowe, zarządzanie i orkiestracja

Contents

1 Introduction **2**

1.1 Goals . . . 2

1.2 Dissertation Thesis . . . 4

1.3 Motivation . . . 4

1.4 Dissertation scope – publication series . . . 6

2 Carrier-grade SDN-based Networks **11**

2.1 Summary of paper [A1] . . . 11

2.1.1 SDN principles . . . 11

2.1.2 Centralized SDN issues . . . 12

2.1.3 SDN scalability improvement methods . . . 12

2.1.4 Proposed scalability model . . . 13

2.1.5 Evaluation results . . . 14

2.2 Summary of research contributions . . . 15

3 Traffic Engineering in Multi-Domain SDN systems **16**

3.1 SDN-TE principles & objectives . . . 16

3.2 Routing optimization – summary of papers [A2][A3] . . . 18

3.2.1 Classical TE issues . . . 18

3.2.2 SDN-TE candidate solutions . . . 19

3.2.3 Proposed Algoritms . . . 20

3.2.4 Hierarchical Deep Reinforcement Learning Load Balancer (HDRL-LB) . . . 20

3.2.4.1 3DQR . . . 22

3.3 Slice Admission Control (SAC) – summary of paper [A4] . . . 24

3.3.1 Role in MANO and TE . . . 24

3.3.2 Time-of-Day-Aware Slice Admission Control (TASAC) . . . 25

3.4 Summary of research contributions . . . 27

4 SDN and SDN-TE applications in 6G Networks **30**

4.1 6G Use Cases– summary of paper [A5] . . . 30

4.1.1 Landscape . . . 30

4.1.2 Advanced Air Mobility (AAM) . . . 30

4.1.2.1 3GPP standardization efforts and research projects . . . 31

4.1.2.2 Provided support and identified gaps . . . 32

4.2 User Plane (UP) Architecture – summary of paper [A6] . . . 32

4.2.1 UP architecture in 5GS & key issues . . . 32

4.2.2 Evolution towards 6G and the role of SDN . . . 33

4.2.3 SDN-UP with DRL-TE (SUP-DTE) 34
4.3 Control Plane Architecture – summary of paper [A7] 35
4.3.1 Core Network evolution directions 35
4.3.2 Hybrid SDN-based Data-Centric Core Network (HSDCN) 36
4.4 Management and Orchestration (MANO) – summary of paper [A8] 37
4.4.1 Evolution of MANO 37
4.4.2 Scalability of orchestration 38
4.5 Summary of the research contributions 39

5 Discussion **42**

6 Summary & Conclusions **46**

7 Academic Achievements **49**
7.1 Publications 49
7.1.1 Journals 49
7.1.2 Conference Proceedings 50
7.1.3 Whitepapers 53
7.2 Conference speeches 53
7.3 Participation in Research Projects 54
7.4 Obtained Funding & Grants: 54
7.5 Other Achievements 55
7.5.1 Contributions to telecommunication standards 55
7.5.2 Research Deliverables 55
7.6 Awards & distincions 56

Appendices **82**

A Publications **82**

B Declarations of co-authors **218**

List of Figures

1.1 Comparison of SDN architectures with deployed Traffic Engineering (TE): classical Centralized SDN (C-SDN) (left) and proposed Hierarchical multi-controller SDN (H-SDN) (right) 6
1.2 Positioning of research actions [R1]–[R8] and articles [A1]– [A8] within 6G research fields related to TE in mobile User Plane (UP). 10

3.1 Generic system modeling and design process applicable to TE process [84] . . . 16
3.2 TE objectives [84] 17
3.3 HDRL-LB entities, interactions and DP view with network switches (circles) and hosts (rectangles) 21
3.4 Internal agent structure and information flow 21
3.5 3DQR entities and interactions (see [A3] for details) 23
3.6 Internal agent structure and information flow (see [A3] for details) 23
3.7 Normalized traffic intensity for a week (obtained from the production network) . 25
3.8 High-level view of TASAC architecture described in [A4] 26

4.1 5th Generation (5G) User Plane Function (UPF) key components [117] 33
4.2 5G UP protocol stack [117] 33
4.3 High-level view of SDN-UP with DRL-TE (SUP-DTE)[A6] 35
4.4 High-level concept of Hybrid SDN-based Data-Centric Core Network (HSDCN) [A7] 37
4.5 Open-Source MANO (OSM) orchestration performance evaluation 39

Chapter 1

Introduction

Since their introduction, the mobile networks have undergone continuous evolution [1]. With the emergence of the 6G concept, they started to face new challenges, including native-Artificial Intelligence (AI) integration, enhanced adaptability, autonomous operations, and interoperability across heterogeneous environments, including Non-Terrestrial Networks (NTNs) [2–6].

One of the key technologies that can support 6G requirements is SDN. The original SDN concept introduces decoupling of network forwarding and control functions. Forwarding devices – Data Plane (DP) – are fully controlled by a single logically centralized SDN Controller (SDNC) – Control Plane (CP). SDNC maintains a global network view, manages all network operations, and exposes NorthBound Interfaces (NBIs), enabling network-to-application adaptation for dynamic software-driven control over the network. In this context, **SDN-TE is key to network performance enhancement**. Its primary objective is to optimize network operation through traffic monitoring and intelligent routing strategies, including methods based on AI, to reach various operational goals.

Despite its potential benefits, **the SDN concept is not scalable**, which makes it inapplicable to carrier-grade networks. Whereas multi-controller SDN architectures present a potential solution to the scalability problem, they dramatically increase the complexity of basic SDN operations and SDN-TE, as well as network deployment costs [7–9].

Nonetheless, the use of SDN in next-generation mobile networks can yield several benefits, including improved adaptability, programmability, advanced network optimization, and implementation of novel network mechanisms. From the application perspective, SDN can serve as a flexible transport network for the mobile core network and facilitate interfacing for Management Plane (MP)-driven optimization.

This dissertation focuses on three crucial aspects of the mobile system evolution towards 6G: **exploitation of SDN in networks**, **scalable SDN-TE algorithms** for intelligent and efficient optimization of users' traffic, and **applications of SDN and SDN-TE in next-generation mobile network architectures**.

1.1 Goals

The primary goal of this dissertation was to design and develop scalable SDN-TE algorithms suitable for deployment in next-generation mobile networks. This objective is structured around three pillars:

- the design and implementation of a **scalable SDN system** enabling fine-grained traffic control operations, which satisfies the generic 6G network requirements;

- the design and implementation of **scalable SDN-TE algorithms**;
- design of **SDN and SDN-TE applications in 6G network** architectures, demonstrating their relevance and benefits for 6G systems.

The dissertation goal has been achieved by:

1. Introduction of the *Hierarchical multi-controller SDN (H-SDN) architecture for traffic management* to address the SDN scalability problem and improve adaptability required by future mobile networks. A queueing-theory-based analytical model was developed to estimate the scalability of single-controller SDN and H-SDN, supporting network planning and dimensioning. Both architectures were implemented in an emulation testbed, and the model was validated experimentally, demonstrating the advantages of H-SDN.

2. Design, implementation, and evaluation of AI-based TE algorithms for scalable and autonomous optimization of network. Three novel algorithms, namely *HDRL-LB*, *3DQR*, and *TASAC*, have been introduced, all leveraging Deep Reinforcement Learning (DRL) to achieve composite operator goals. *HDRL-LB* and *3DQR* focus on routing optimization in H-SDN, optimizing objectives such as load distribution, total throughput, or revenue, in terrestrial and Three-dimensional (3D) networks (comprising connected terrestrial, aerial, and satellite segments), respectively. The methods enable concurrent local and global optimization, which is often omitted in State of the Art (SoA) and crucial in the context of multi-network 6G environments. The third algorithm, *TASAC*, implements a complementary admission control mechanism that exploits daily user activity patterns to optimize resource utilization.

3. Proposal of metrics abstractions for H-SDN to improve scalability of SDN-TE. The approach reduces complex domain topology and metrics to a simple overlay graph composed of domain gateways, virtual links between them, and inter-domain links. Metric abstraction in H-SDN has been insufficiently addressed in the state of the art. The proposed metrics were successfully utilized by *HDRL-LB* and *3DQR*, demonstrating their effectiveness for global optimization.

4. Analysis of the 6G landscape, including ongoing research activities, standardization efforts, and key vertical sectors. The study has focused on Advanced Air Mobility (AAM) use cases due to high requirements for network flexibility. The review serves as a foundation for designing 6G network architectures based on H-SDN capable of accommodating SDN-TE solutions.

5. Design of 6G network architectures leveraging SDN and SDN-TE to address key challenges of emerging 6G frameworks, including reuse of 5G achievements, improved adaptability, and sustainable evolution. Two architectures are proposed: *SUP-DTE*, a user-centric UP

framework with integrated intelligent SDN-TE, and *HSDCN*, a data-centric mobile core architecture using content-based routing for improved customizability and ability to introduce next-generation functionalities. Compared to SoA approaches, both solutions address critical economic constraints by enabling continuous evolution and incremental integration of 6G mechanisms while maintaining compliance with the 3GPP 5G architecture. To complement the proposed architectures, the performance and scalability of a popular orchestrator, OSM, have been analyzed to emphasize the importance of scalability in carrier-grade network processes.

1.2 Dissertation Thesis

The research thesis investigated in this dissertation is formulated as follows:

Proposed Hierarchical multi-controller SDN (H-SDN) architecture and SDN-TE algorithms enable scalable, adaptive, and efficient traffic control suitable for next-generation mobile networks.

Its validity is demonstrated through the series of papers listed in Section 1.4, summarized in Chapters 2, 3, and 4, and jointly discussed in Chapter 5.

1.3 Motivation

The motivation for this dissertation stems from the rapid evolution and increasing complexity of next-generation mobile systems, particularly in the context of the mobile User Plane (UP). The design shift introduced in 5G System (5GS), which emphasized support for vertical industries and specialized services, has significantly increased demand for network flexibility, adaptability, and dedicated support to meet service requirements [10].

This trend is expected to intensify in next-generation mobile networks [11, 12] due to the much broader range of supported use cases, integration of heterogeneous networks including Non-Terrestrial Network (NTN) systems (i.e., Network of Networks concept), and native support for building and operating AI solutions, including, i.a., data collection, processing, exposure, or implementing AI model training pipelines [10]. Consequently, traffic patterns in the network are expected to become increasingly complex, exhibiting hidden dependencies and behaviors that will require intelligent and automated optimization mechanisms to ensure efficient network operation.

The limited flexibility of network control in current systems has already been recognized by 3rd Generation Partnership Project (3GPP), which has initiated studies to enhance network adaptability in 6G System (6GS) [6, 12, 13]. The major limitation is the use of stateful overlay tunnels to convey user traffic in the network by means of GPRS Tunneling Protocol (GTP). Its primary concerns are the large signaling overhead for every tunnel reconfiguration, limited routing flexibility, and the constrained TE and mobility at only one end of the path. The latter is critical in scenarios requiring concurrent network path reconfiguration at both the network and user sides, e.g., due to joint user–application mobility or infrastructure mobility in NTN. To address this issue, the 3GPP has begun studies to include Segment Routing over IPv6 (SRv6) protocol as an underlying protocol for GPRS Tunneling Protocol (GTP) or as a GTP replacement [12]. While this solution can improve network programmability, it will generate significant data overhead due to the encoding of network path information and operations within packet headers [14].

In this context, the use of SDN can bring multiple benefits:

- **Programmable and flexible network control** – enabling dynamic traffic management and adaptation to current network conditions; The application-driven control enables implementation of advanced network mechanisms such as TE, Quality of Service (QoS)-control, security mechanisms, etc.

- **Global network visibility** – facilitating network-wide operations and optimization. Moreover, network contextual information can be easily incorporated to improve decision-making and align with the strategic goals of the network operator.

- **Flexible extension** – network and technology abstraction provided by SDN controllers enables the development of evolving network platforms. This is crucial in the 6G context, where SDN-native protocols can be used not only to implement solutions proposed by 3GPP such as SRv6 [15], or the GTP protocol [16], but to fully replace them. Moreover, SDN enables the implementation of advanced traffic steering mechanisms, such as label-based forwarding, multi-path routing, or content-based routing, which are well-suited to cloud environments.

So far, however, the SDN paradigm has not achieved widespread commercial success, due to the lack of scalability (discussed further in Chapter 2 and in [A1]). Today, SDN is predominantly deployed in data center networks (for virtualization, tenant isolation, and load balancing), enterprise Software-Defined Wide Area Networks (SD-WANs), or Software-Defined Internet Exchange Points (SDXs) (optimizing inter-domain traffic) [17]. The latter two implement SDN-inspired centralized control or hybrid networks rather than “pure” OpenFlow (OF)-based SDN [18]. To date, the only notable large-scale SDN deployments include Google’s B4 backbone [19], Microsoft SWAN [20], and selected research testbeds [21]. Also, SDN adoption in operator mobile networks remains limited in practice.

To address scalability concerns, several approaches can be adopted, of which the most popular is adopting multi-controller architectures [22, 23]. In this thesis, the use of Hierarchical multi-controller SDN (H-SDN) architecture, depicted in Fig. 1.1, is proposed. Despite improved scalability, the hierarchy introduces several open research problems, which include:

- efficient **coordination of distributed CP operations** to ensure consistent End-to-End (E2E) traffic management and reliable QoS provisioning across multiple, heterogeneous domains,

- **reduction of CP and monitoring overhead** caused by mechanisms used to maintain the global visibility over the large-scale multi-controller environment. These include topology discovery, monitoring data collection, and controller state synchronization.

- **intelligent and scalable TE**, capable of jointly optimizing domain- and global-level goals, while preserving overall network stability,

- design of **lightweight, autonomous, and adaptive TE solutions** suitable for highly complex and dynamically changing environments, such as NTNs.

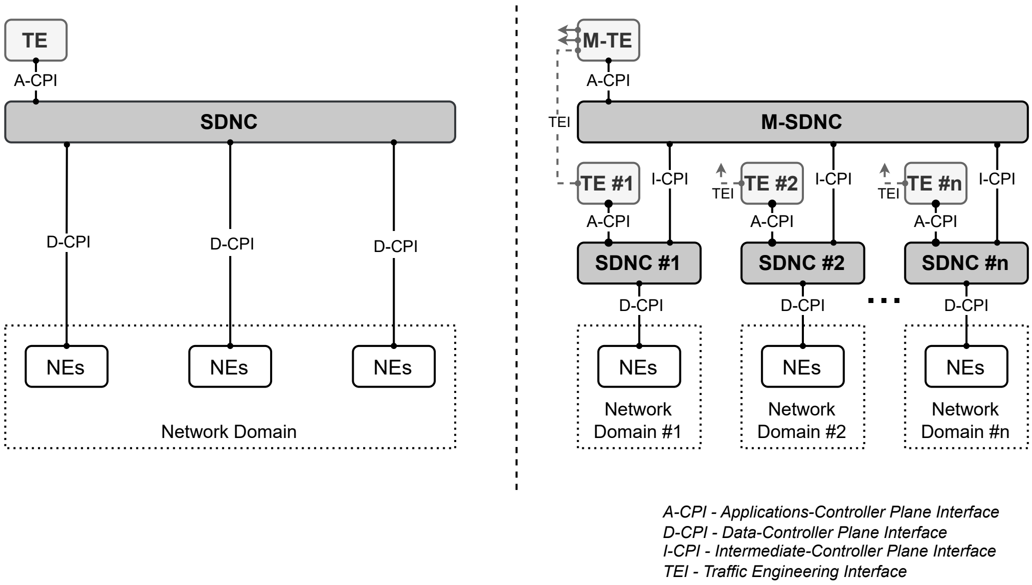


Figure 1.1: Comparison of SDN architectures with deployed TE: classical C-SDN (left) and proposed Hierarchical multi-controller SDN (H-SDN) (right)

The developed SDN and SDN-TE mechanisms, described in this work, aim to address the above problems, while considering the general 6G network landscape.

1.4 Dissertation scope – publication series

The objectives of this dissertation were achieved through the author's theoretical and practical research activities on various aspects of SDN, SDN-TE, and 6G networks. The dissertation is based on a series of 8 peer-reviewed publications:

[A1] **Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Evaluating Scalability of Hierarchical Multi-controller SDN: A Holistic Approach". In: *Journal of Network and Systems Management* 34.4 (May 2026), p. 107. ISSN: 1573-7705. DOI: `10.1007/s10922-026-10087-w`** – p. 84
Type of publication: **journal article**
Impact Factor: **3.6**;
Ministerial points (08.05.2026): **70**

Contribution of the thesis author: **80%**

[A2] **Robert Kołakowski, Slawomir Kukliński, and Lechosław Tomaszewski. "Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Networks". In: *2024 IFIP Networking Conference (IFIP Networking)*. June 2024, pp. 607–612. DOI: `10.23919/IFIPNetworking62109.2024.10619899`** – p. 126
Type of publication: **conference materials**
Impact Factor: **–**;
Ministerial points (08.05.2026): **70**
Contribution of the thesis author: **80%**

[A3] **Robert Kołakowski et al. "Hierarchical Traffic Engineering in 3D Networks Using QoS-Aware Graph-Based Deep Reinforcement Learning". In: *Electronics* 14.5 (Mar. 2025), p. 1045. ISSN: 2079-9292. DOI: `10.3390/electronics14051045`** – p. 132
Type of publication: **journal article**
Impact Factor: **2.6**;
Ministerial points (08.05.2026): **100**
Contribution of the thesis author: **75%**

[A4] **Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Time-of-Day-Aware Slice Admission Control". In: *2023 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. Sept. 2023, pp. 199–204. DOI: `10.1109/MeditCom58224.2023.10266645`** – p. 163
Type of publication: **conference materials**
Impact Factor: **–**;
Ministerial points (08.05.2026): **20**
Contribution of the thesis author: **60%**

[A5] **Lechosław Tomaszewski and Robert Kołakowski. "Advanced Air Mobility and Evolution of Mobile Networks". In: *Drones* 7.9 (Sept. 2023), p. 556. ISSN: 2504-446X. DOI: `10.3390/drones7090556`. (Visited on 02/27/2025)** – p. 169
Type of publication: **journal article**

Impact Factor: **4.4**;
Ministerial points (08.05.2026): **40**
Contribution of the thesis author: **35%**

[A6] **Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "On 6G-Enabled SDN-Based Mobile Network User Plane with DRL-Based Traffic Engineering". In: *Artificial Intelligence Applications and Innovations. AIAI 2024 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2024, pp. 124–138. ISBN: 978-3-031-63227-3. DOI:** `10.1007/978-3-031-63227-3_9` – p. 193
Type of publication: **conference materials**
Impact Factor: **–**;
Ministerial points (08.05.2026): **20**
Contribution of the thesis author: **80%**

[A7] **Robert Kołakowski, Thierry Lejkin, and Victorien Romain. "Hybrid SDN-Based Data-Centric Mobile Core Network with Content-Based Routing". In: *2025 International Conference on 6G Networking (6GNet 2025)*. Paris, France, Dec. 2025, pp. 126–130. DOI:** `10.1109/6GNet68413.2025.11314080` – p. 207
Type of publication: **conference materials**
Impact Factor: **–**;
Ministerial points (08.05.2026): **20**
Contribution of the thesis author: **80%**

[A8] **Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "Performance evaluation of the OSM orchestrator". In: *2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. Nov. 2021, pp. 15–20. DOI:** `10.1109/NFV-SDN53031.2021.9665052` – p. 212
Type of publication: **conference materials**
Impact Factor: **–**;
Ministerial points (08.05.2026): **20**
Contribution of the thesis author: **55%**

The series of publications forms a sequence of research and development activities spanning the fields of SDN, SDN-TE, and 6G network architectures. The dissertation begins with an investigation of the SDN scalability problem. Paper [A1] introduces analytical models to evaluate the scalability of the single-controller and H-SDN architectures. The results indicate the superior scalability of H-SDN, motivating its adoption as the foundation for the SDN-TE algorithms. Building on these findings, papers [A2] and [A3] present designed SDN-TE algorithms for routing optimization in H-SDN networks, targeting two deployment scenarios: Terrestrial Networks (TNs) and integrated 3D networks. Proposed solutions are evaluated using emulation and simulation testbeds described in [A2] and [A3], respectively. Paper [A4] proposes a complementary TE mechanisms – Slice Admission Control (SAC) – which acts as an entry-level control of total resource consumption in the network. In a representative deployment, Slice Admission Control (SAC) [A4] decides whether to admit a slice (a group of users) into the network, and algorithms shown in [A2], [A3] optimize incoming traffic.

Papers [A5]-[A8] focus on applications of SDN and SDN-TE in future mobile networks. Paper [A5] explores the requirements of the Advanced Air Mobility (AAM) vertical sector, which features stringent flexibility and performance demands. These motivate the adoption of SDN. Papers [A6] and [A7], present H-SDN-based 6G-ready architectures for mobile UP and CP capable of running SDN-TE and promoting sustainable evolution of the network. Finally, paper [A8] presents complementary work on performance and scalability evaluation of the most popular standardization-compliant orchestrator.

To reach the dissertation goal, the following research actions have been taken:

- **[R1]** Synthesis of knowledge concerning the scalability of the SDN Control Plane and gap identification.
- **[R2]** Synthesis of knowledge concerning network evolution towards 6G, including standardization, emerging 6G use cases, requirements, architectures, and mechanisms.
- **[R3]** Design of SDN CP scalability model and its validation in emulation environment.
- **[R4]** Design of 6G Network architectures leveraging SDN as its foundation for efficient Traffic Engineering.
- **[R5]** Design and development of SDN emulation testbed and event-driven SDN simulator for benchmarking SDN architecture and TE algorithms.
- **[R6]** Design, development, and performance evaluation of AI-driven Traffic Engineering algorithms targeting H-SDN networks.
- **[R7]** Development of a complementary Traffic Engineering mechanism, focused on admission control, and evaluation using simulations.

[R8] Deployment of open-source network orchestrators, development of monitoring instrumentation for performance evaluation and benchmarking.

Fig. 1.2 presents the main topics explored in the dissertation and illustrates relationships across 6G research areas related to UP optimization, and maps the undertaken research actions and articles. Table 1.1 summarizes the contribution of all papers [A1]-[A8] towards research actions [R1]-[R8].

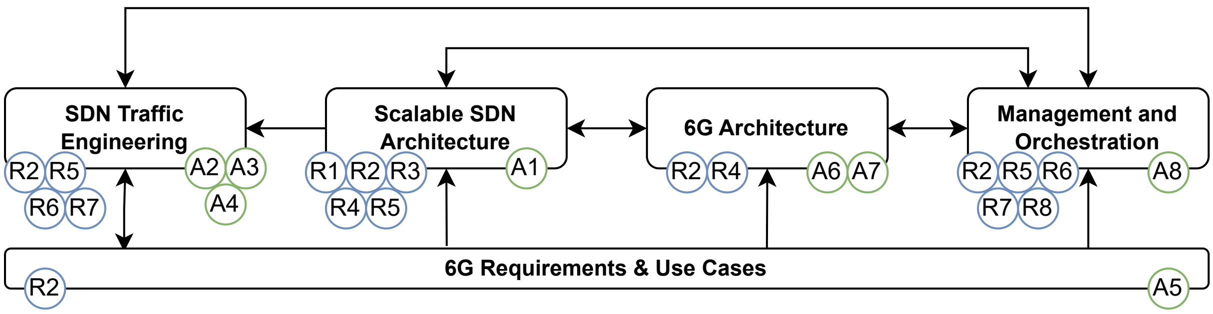


Figure 1.2: Positioning of research actions [R1]–[R8] and articles [A1]– [A8] within 6G research fields related to TE in mobile UP.

Article	Action	Chapter
[A1]	[R1] [R3] [R4] [R5]	Chapter 2, Section 2.1
[A2] [A3]	[R2] [R5] [R6] [R2] [R5] [R6]	Chapter 3, Section 3.2
[A4]	[R2] [R7]	Chapter 3, Section 3.3
[A5]	[R2]	Chapter 4, Section 4.1
[A6]	[R2] [R4]	Chapter 4, Section 4.2
[A7]	[R2] [R4]	Chapter 4, Section 4.3
[A8]	[R2] [R8]	Chapter 4, Section 4.4

Table 1.1: Implementation of research actions [R1]–[R8] leading to the achievement of the dissertation goals in articles [A1]–[A8].

Chapter 2

Carrier-grade SDN-based Networks

2.1 Summary of paper [A1]

2.1.1 SDN principles

Traditional Internet Protocol (IP) networks rely on distributed routing protocols for packet forwarding. While scalable, in large networks, they suffer from slow convergence [32] and limited support for dynamic forwarding control. Software-Defined Networking (SDN) concept addresses these issues. Compared to IP networks, SDN decouples traffic forwarding and control, enabling network-to-application adaptation. The original SDN concept, later referred to as Centralized SDN (C-SDN), introduces a single SDN Controller (SDNC) – SDN Control Plane (CP) – that manages forwarding devices (switches) – SDN DP – and provides a network abstraction for higher level applications (routing, TE, policing, QoS-control, etc.) – SDN Application Plane (AP). In C-SDN, the SDNC has a global view of the network (topology, link and node state, etc.), which simplifies management and enables network-wide optimization. Moreover, SDN enables the reuse of commodity hardware for network automation, thereby reducing capital expenses for DP infrastructure. In SDN, traffic is represented using a flow abstraction, which refers to a set of packets sharing common header attributes and treated as a single forwarding unit by the network. Overall, SDNC can operate in three modes:

- proactive – forwarding rules are installed before the flow arrives in the network,
- reactive – forwarding rules are installed dynamically, when the flow arrives in the network,
- hybrid – combining proactive and reactive modes.

For network operators, the reactive mode offers the highest benefits as it addresses two key practical limitations:

- **flow table capacity of hardware switches** – typically implemented using Ternary Content-Addressable Memory (TCAM), which is costly and limited in size [33]. As a result, the proactive installation of fine-grained flow rules is often infeasible in production scenarios. Reactive operation mitigates this issue by installing forwarding rules on demand and removing the ones for inactive flows to conserve TCAM.
- **simplified management and optimized routing** – routing decisions can be made dynamically based on the current network state, which minimizes the need for rerouting flows, which is associated with costs. Moreover, the configuration management can be simplified as flows can be handled dynamically without proactive routing management.

2.1.2 Centralized SDN issues

Despite many benefits, however, the C-SDN concept suffers from several issues, where the most vital are [8, 34–36]:

- **SDNC placement** – the physical location of SDNC, should minimize Round-Trip Time (RTT) between SDNC and switches for efficient operations. Also, the delays introduced by physical distance must be compensated for by the controller to ensure synchronized installation of forwarding rules.
- **Reliability** – single SDNC constitutes a single point of failure, thus requiring extensive protection and redundancy to prevent global network outages.
- **CP scalability** – referring to a system's ability to maintain efficient operation as flow arrival rates increase. A single centralized controller can become a bottleneck in large or highly dynamic networks, significantly affecting reaction times and overall network performance.

The issue of CP scalability is inherent for the reactive operation mode of SDN, as SDNC must process control messages for every new flow in the network. In most common OF-based SDN networks [18], scalability is further exacerbated by monitoring data collection [37–39] and network topology discovery (e.g., via OpenFlow Discovery Protocol (OFDP)) [40, 41], as all operations are carried over the CP channel by SDNC. Moreover, as OF lacks built-in QoS support, in mobile networks SDNC should host extensions for QoS provisioning (e.g., for latency measurements), which will further aggravate the issue and increase CP load. The SDN CP scalability has remained a key bottleneck limiting wide-scale adoption of SDN in carrier-grade networks [42–46]. Therefore, improving scalability has become a central research area in SDN [42–44, 47–49].

2.1.3 SDN scalability improvement methods

The SDN CP scalability issue remains a multi-dimensional challenge that requires careful assessment during the network design phase [9]. Existing mitigation strategies can be grouped into three categories:

1. **Distributed CP architectures** [22, 43, 44, 48–55];
2. **CP performance enhancements** via parallelized processing in SDNCs [56–59], switch reassignment across controllers [60, 61], combining proactive and reactive operation [42], optimizing network monitoring [43, 62, 63], optimized controller placement (in terrestrial [64–66] and satellite SDN environments [61, 67]), etc.
3. **DP improvements** in terms of functionality and protocol efficiency to offload SDNC, e.g., P4-based TE algorithms [68, 69], In-band Network Telemetry [70], or QoS routing [71].

The vital strategy to handle the scalability problem involves deploying multi-controller SDN architectures to distribute network control functions across multiple cooperating controllers organized in flat (e.g., ONIX, DISCO, ELASTICON [51–53]) or hierarchical frameworks (Terraflow SDN, Orion, Kandoo [54, 55, 72]). Multi-controller SDN architectures, however, must continuously exchange information to coordinate control operations and maintain a consistent global network state [42]. This incurs significant overhead and inter-controller communication costs, which increase CP load and degrade controller performance [73]. Multi-controller architectures are also more complex, complicating deployment and management in operational scenarios.

The SDN CP scalability is often assessed using isoefficiency (i.e., the rate at which the problem size must scale with increasing processor count to maintain constant parallel efficiency) and isospeed (the rate at which the problem size must grow with the number of processors to maintain constant execution time) [44]. The CP scalability is also closely linked to CP performance metrics. The most important in this context are controller utilization (the consumed capacity of SDNC to handle new flows), Time-To-Serve (TTS) (the time to install the new flow since its arrival at the ingress switch), and physical resource consumption (CPU, memory, storage, and disk operations), indicating the cost to deploy the SDN architecture variant.

Prior work on SDN CP modeling indicates that distributed architectures can offer improved scalability compared to centralized ones [47–49, 74, 75]. A key limitation of these proposals is that they often assess performance or deployment costs without accounting for the overhead of monitoring and topology-discovery mechanisms [43]. Additionally, the SoA approaches predominantly rely on simulations and neglect the impact of CP distribution and transient effects on the overall performance.

2.1.4 Proposed scalability model

Paper [A1] proposes a unified scalability model for C-SDN and H-SDN architectures operating in reactive mode. The model jointly considers CP performance and the deployment's resource cost while accounting for traffic overhead from custom protocols (e.g., monitoring and topology discovery). The model is based on the Flatto–Hahn–Wright (FHW) queueing model [76], in which a single system call is split across several servers (fork), and final success depends on completing the call at each server (join). This behavior is analogous to the reactive operation of H-SDN, in which domains are managed by their own SDNCs that are subjected to an upper-layer controller Hierarchical SDN Controller (HSDNC). HSDNC manages E2E operations using an abstracted view of the entire network in which topology is simplified to domain gateway nodes, inter-domain links, abstract intra-domain links (representing connectivity within a domain between two gateways), and hosts that issue/receive traffic. Each incoming flow is either processed by SDNC (intra-domain routing) or by HSDNC (inter-domain routing), which, after establishing a rough path composed of gateways and inter-domain links, delegates intra-domain routing and path instantiation to selected SDNCs.

Scalability is assessed from two perspectives:

- CP performance – controller utilization and Time-To-Serve (TTS) – a measure of time needed by the controller to establish a path and install forwarding rules for the flow.
- Resource consumption – compute, memory, storage, disk, and communication resources required by the SDN CP.

The above metrics allow comparison of different SDN architectures. The overall scalability is calculated using an introduced scalability index.

2.1.5 Evaluation results

Paper [A1] validates the analytic model experimentally using an emulator built on open-source SDN tools with Ryu SDN controller [59], Mininet network emulator [77], and Open Virtual Switch (OVS) switches [78] as DP emulation. The study evaluates compute, storage, and connectivity requirements of SDN CP components and demonstrates the scalability benefits and resource trade-offs of distributed SDN control architectures. The paper focuses on reactive-mode SDN operation and compares C-SDN with H-SDN. The focus is placed on C-SDN, and H-SDN due to their growing relevance to 6G architectures [79–82] and their ability to provide global control capabilities. In contrast, distributed flat architectures are omitted, as they suffer from lower performance, increased coordination overhead, and limited network visibility [50], which makes QoS enforcement—critical in 6G—challenging in large-scale scenarios [45].

Evaluation results prove superiority in terms of CP scalability of H-SDN variant compared to C-SDN, especially in high-load and in large networks. Adoption of H-SDN significantly reduces TTS up to 55% in terms of intra-domain path setup. For inter-domain paths, the overhead is significant under low load (up to a 90% increase in TTS), but it diminishes at scale and becomes negligible in medium- and large-scale SDN networks. Regarding resource consumption, CPU usage exhibits sublinear growth, memory scales linearly, while bandwidth and disk I/O show nonlinear growth, with bandwidth increasing by up to eight times compared to C-SDN as the number of SDNCs and load increase. Overall, distributed CP increased CP capacity and avoids single-controller bottlenecks inherent to C-SDN. Finally, the scalability gains depend on a balanced workload distribution, as congestion in a single domain can degrade both intra- and inter-domain performance. Therefore, the network design and controller placement phases are critical for deploying carrier-grade SDN networks and for using them as the basis for mobile network User Plane (UP).

The findings of the paper [A1] served as the basis for the architectural and design choices for architectures presented in papers [A6] and [A7], as well as TE algorithms [A2] and [A3]. Each of the contributions uses hierarchical multi-controller SDN as its basis for traffic control flexibility and scalable operations.

2.2 Summary of research contributions

The key contributions of the thesis author presented in the paper **[A1]** are as follows:

- Analysis and synthesis of existing knowledge related to the architecture, measurement, modeling, and challenges associated with the scalability of the SDN CP, along with the identification of research gaps.
- Development of a comprehensive analytical framework for evaluating Control Plane (CP) scalability in SDN, covering both Centralized SDN (C-SDN) and Hierarchical multi-controller SDN (H-SDN) architectures. Unlike prior works, the model jointly evaluates controller utilization, Time-To-Serve (TTS) for flow setup, and infrastructure resources needed to deploy SDN architectural variant,
- Introduction of mathematical scalability models based on Flatto–Hahn–Wright (FHW) queuing (M/M/1 and fork-join structures) to capture SDN controller utilization, load distribution, and stability conditions under increasing traffic demands. The framework enables analytical estimation of controller load, saturation points, and performance degradation for specific traffic demand.
- Definition and formalization of key SDN scalability performance metrics, including controller utilization and Time-To-Serve (TTS), incorporating both user-generated flow traffic and CP operational overhead (e.g., OpenFlow-based monitoring, topology discovery) omitted in prior works. Furthermore, a resource consumption model for the SDN CP, covering compute, memory, disk I/O, and bandwidth utilization, is introduced based on controller utilization and overall system load.
- Design and implementation of a realistic SDN emulation testbed based on open-source tools (i.a., Mininet, OVS, Ryu) and instrumentation for scalability measurements, and automated tests execution.
- Proposal of metric for CP scalability estimation and quantitative demonstration of H-SDN architectures scalability gains compared to traditional C-SDN approach.
- Empirical validation of the proposed models demonstrating strong agreement between analytical and experimental results.

Chapter 3

Traffic Engineering in Multi-Domain SDN systems

3.1 SDN-TE principles & objectives

Network TE is a practice of design, evaluation, and optimization of operational networks to reach specific operators' objectives [83]. TE encompasses the application of technology and scientific principles to measure, characterize, model, and control data flow in the network with the major goal of enhancing its performance (cf. Figure 3.1).

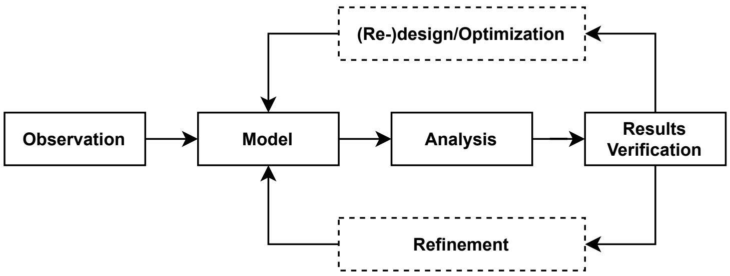


Figure 3.1: Generic system modeling and design process applicable to TE process [84]

The TE process typically begins with monitoring of network behavior and traffic characteristics to construct an abstract system model. Subsequently, an iterative optimization process is performed, consisting of: i) consecutive tests and refinement of the system model based on performance metrics (lower loop) and ii) iterative system dimensioning and optimization or re-design (upper loop). Given the dynamic nature of modern networks, TE can be viewed as a long-term process spanning the entire network lifecycle.

Classical TE considered long-horizon tasks such as capacity planning, congestion avoidance, or general robustness enhancement, whereas real-time network adaptation capabilities remained limited [85]. The programmability, centralized control, and global network visibility introduced by SDN (cf. Chapter 2) enable the development of more dynamic TE mechanisms that can adapt network operation to changing conditions in real time. Broadly, the objectives of SDN-TE can be grouped into following categories [85–87](3.2):

- **Network performance** – concerning optimization of network-level Performance Indicators (PI) such as minimizing E2E latency, packet loss and jitter, maximizing bandwidth, and congestion avoidance. These objectives are commonly achieved through intelligent routing, admission control, traffic distribution across the network infrastructure, network resource reallocation, prioritization, or limiting access to congested resources.

- **Resource Utilization** – concerning minimization of the usage of physical/virtualized network, compute, and storage resources to optimize costs or energy consumption, which can be achieved by traffic redirection, deactivation of hardware/software resources, adaptation to well-known data transfers to the least-busy moments (e.g., database migrations/synchronization during night hours), etc.

- **Service Performance** – referring to higher-level objectives perceived by the user, and including maximizing QoS, Quality of Experience (QoE), maintenance of Service Level Agreement (SLA)/user intent, and fair resource distribution across users, while maintaining the benefit from allocating resources (utility).

- **Cross-cutting/System-level objectives** – comprising all other objectives that are related to system-level functionalities such as reliability & resilience (e.g., failover mechanisms, improving fault tolerance and stability, employing mechanisms for better service continuity), security (malicious traffic redirection, Deep Packet Inspection (DPI), access control, traffic scrubbing), SDN efficiency (e.g., optimizing CP load, CP performance, scalability), multi-tenancy optimization (e.g., resource partitioning, isolation, resource fairness) or cost (e.g., energy pricing, costs of compute/network lease, etc.).

It should be emphasized that the objectives listed above are strongly interrelated and often represent compositions of multiple lower-level objectives. For example, QoS optimization typically involves simultaneously optimizing several network performance metrics, such as delay, bandwidth, packet loss, and packet error rate.

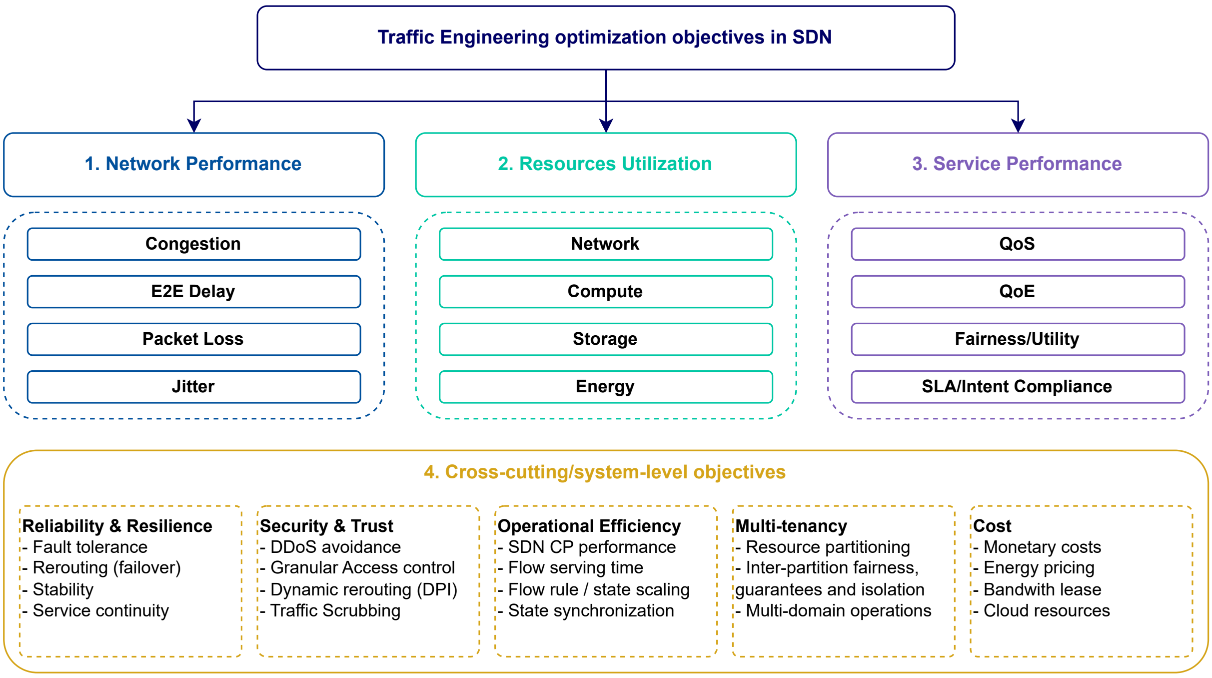


Figure 3.2: TE objectives [84]

This section focuses on research activities related to Traffic Engineering (TE) in hierarchical multi-domain SDN-based networks, due to scalability benefits discussed in Chapter 2 and paper [A1]. Paper [A4] presents the SAC algorithm that is complementary to TE algorithms presented in papers [A2] and [A3].

3.2 Routing optimization – summary of papers [A2][A3]

3.2.1 Classical TE issues

Future mobile networks will require highly adaptive and autonomous TE mechanisms due to the increased variety of services, operation across heterogeneous infrastructures, and extending the network beyond the terrestrial segment due to close integration with Non-Terrestrial Network (NTN) (i.e., aerial and satellite networks)[88–90] – envisioned to achieve ubiquitous global broadband access[91]. In the context of the satellite systems, the Mobile Network Operators (MNOs) are interested mainly in integrating Low Earth Orbit (LEO) constellations, due to acceptable transmission latencies (30-50 ms) [92]. LEO networks, however, introduce several challenges due to their high mobility, resulting in frequent topology changes [93], and limited capacity of optical Inter-Satellite Links (ISLs) and Feeder Links (FLs) (links between satellite nodes and terrestrial gateways). To preserve service continuity and use NTN resources efficiently, intelligent TE supporting QoS-aware routing, E2E cross-network operations, and load balancing, is vital.

Classical TE approaches, such as Open Shortest Path First (OSPF) [94], Multi-Protocol Label Switching Traffic Engineering (MPLS-TE), and classical optimization methods (e.g., Integer Linear Programming (ILP)), were originally designed to optimize legacy Terrestrial Network (TN) only, which feature limited topology and traffic dynamics. To this end, classical methods face the following limitations [89, 92, 95]:

- **Adaptability to dynamic environments** – classical TE methods require periodic exchange of information regarding topology and link-state across the network devices, commonly combined with static optimization rules. In environments with rapidly changing topologies, such as NTN systems, these methods result in slow routing convergence, instability in the network, and excessive signaling overhead [96]

- **Intelligent optimization logic** – traditional TE mechanisms rely on predefined metrics such as link cost or utilization and manually engineered optimization rules (e.g., shortest paths, static link weights, fixed constraints) for network optimization. As a result, their ability to capture complex traffic dynamics and incorporate predicted future conditions or contextual information is limited, leading to suboptimal routing decisions. This issue is especially critical in the context of NTN, which frequently undergoes topology changes, including link breaks, link quality degradation, and network node isolation, making link prediction and fast path updates essential (via temporal and predictive routing). This limits their ability to provide resilient QoS guarantees in highly dynamic environments.

- **Network heterogeneity** – future mobile networks will combine diverse TN and NTN systems into a unified heterogeneous network, creating the so-called “Network of Networks” [10]. Such an environment will combine different protocols, technologies, control mechanisms, and service capabilities, requiring intelligent, scalable, and automated TE adaptable to domain-level characteristics and administrative constraints. To this end, the flexibility of conventional TE methods and the inclusion of non-network data into decision processes are limited.

- **Resource utilization** – conventional TE commonly relies on shortest-path and metric-based routing methods, which tend to concentrate traffic on currently optimal paths. Due to the timely update of routing state, transient network changes are often not addressed, leading to congestion, reduced throughput, and suboptimal traffic distribution.

- **Resilience** – conventional routing approaches have limited capability to proactively mitigate failures, intermittent links, or topology fragmentation common in NTN segment. Common rerouting events may result in service degradation, packet loss, and SLA violations. Possible solutions include multi-path routing [97], store-and-forward buffering [98, 99], and predictive approaches.

3.2.2 SDN-TE candidate solutions

The above limitations motivate a transition to AI-driven TE, in which routing decisions are continuously adapted based on observed network state. SDN plays a special role in enabling such optimization through its inherent programmability, which allows deploying TE applications that optimize the network in real time via SDN NBIs. At the same time, classical C-SDN architectures suffer from scalability limitations (cf. [A1]), especially when network size and traffic dynamics increase. While distributed SDN architectures improve scalability [100], they largely complicate E2E TE due to each controller’s partial visibility into the global network state and the need to coordinate TE actions across controllers, which can also affect system scalability.

AI/Machine Learning (ML) methods are a natural candidate for TE due to their ability to graph hidden relationships between different processes concurrently occurring within the mobile systems. To this end, Deep Reinforcement Learning (DRL) is one of the fundamental AI-based optimization methods, which is commonly applied for optimizing C-SDN networks [23, 101–104]. Although DRL is widely considered for TE in TNs because of its adaptability and strong performance, its application in NTNs remains difficult because learned policies depend heavily on environments, convergence degrades in time-varying settings, and observability and scalability challenges remain unresolved, especially in distributed scenarios [105–108]. Moreover, while several Multi-Agent Deep Reinforcement Learning (MADRL) approaches have emerged as promising alternatives [105–107, 109–112], enabling more efficient distributed optimization, there remains a lack of approaches that jointly support multi-level optimization (domain- and global-level) while accounting for E2E QoS constraints, which are critical in operator networks.

3.2.3 Proposed Algoritms

Papers [A2] and [A3] propose two novel algorithms for autonomous AI-based TE targeting optimization of user traffic in next-generation mobile networks. Both algorithms target hierarchical multi-controller SDN deployments and introduce complementary approaches that use Multi-Agent Deep Reinforcement Learning (MADRL) to implement intelligent routing at the local and global levels.

3.2.4 Hierarchical Deep Reinforcement Learning Load Balancer (HDRL-LB)

Principles Paper [A2] proposes HDRL-LB, a multi-agent Deep Deterministic Policy Gradient (DDPG)-based routing algorithm [113] enhanced with a Double Deep Q-Network (DDQN)-based critic [114], which optimizes traffic distribution in the network and aggregate throughput, representing the total traffic successfully carried by the network corresponding to MNO's profits. Hierarchical Deep Reinforcement Learning Load Balancer (HDRL-LB) is designed for hierarchical multi-domain SDN (cf. Fig. 3.4), where:

- Domain SDNCs handle intra-domain control, a Global SDNC (GSDNC) coordinates E2E routing,
- Domain Load Balancing Agents (DLBAs) optimize local domains, and Global Load Balancing Agent (GLBA) optimizes the network globally.

The key design idea is to combine fast local optimization with slower overlay global one, so that domain controllers can quickly adapt the network to local traffic changes while the global controller manages traffic distribution across domains to stabilize and optimize the network E2E. This avoids the issue where local optimization alone may cause congestion on inter-domain links or fail to achieve the global optimum. To improve scalability and support hybrid or multi-provider environments, HDRL-LB introduces network abstractions. The GSDNC does not see full domain internals; instead, it operates on an abstracted overlay graph composed of Border Nodess (BNs), hosts, inter-domain links, and abstracted intra-domain links between BN pairs. To this end, the E2E routing by GSDNC (for inter-domain flows) is realized by computing network paths in the overlay graph and delegating intra-domain path computation and enforcement to domain SDNCs. The network paths on the domain level and the global level are computed with the shortest path algorithm over the routing graphs provided by DRL agents (DLBAs, GLBA).

To represent the properties of the overlay graph, paper [A2] introduces specialized metrics which include capacity (calculated using max-flow algorithm), utilization (calculated by subtracting the current capacity under traffic). In addition to link-state metrics (bandwidth, capacity), the algorithm considers normalized time to enable the agents to learn recurring traffic pat-

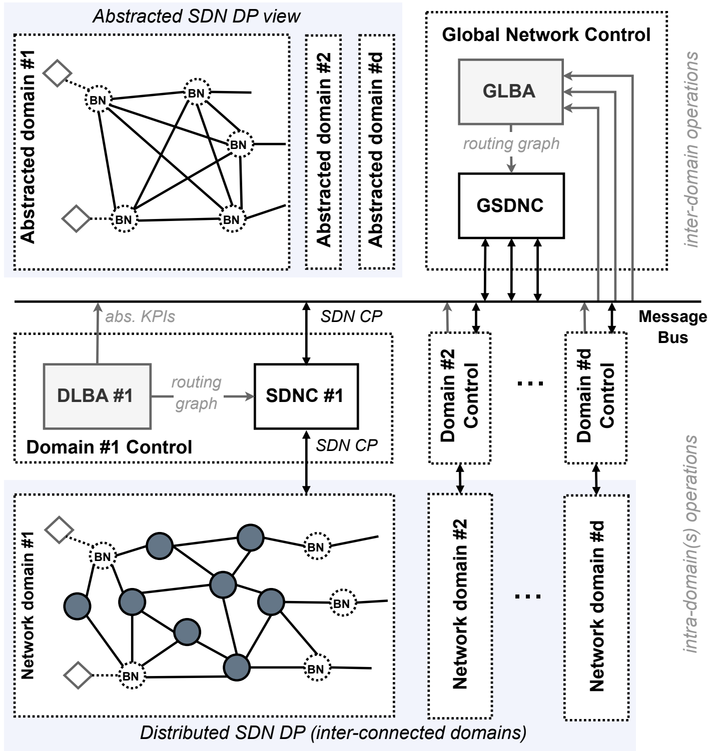


Figure 3.3: HDRL-LB entities, interactions and DP view with network switches (circles) and hosts (rectangles)

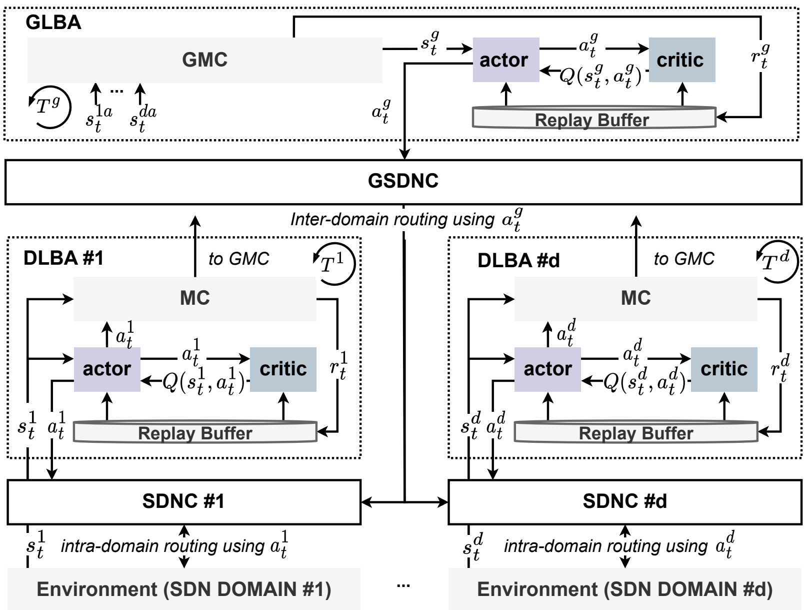


Figure 3.4: Internal agent structure and information flow

terns across time (as proposed in [A4]). Moreover, a set of composite metrics is defined to evaluate load distribution (domain, abstracted domain, and global load-balancing factors) and a measure of aggregate throughput.

This approach reduces monitoring overhead and control complexity by having domain-level SDN controllers exchange aggregated monitoring data (composite graph metrics) with GSDNC instead of fine-grained per-link information, while still supporting E2E routing. Moreover, HDRL-LB algorithm considers the SDN CP overhead by performing routing graphs updates at fixed intervals (frequent updates by DLBAs, slow updates by GLBA to stabilize the network).

Evaluation & Results The HDRL-LB algorithm was evaluated against the baseline Weighted Dijkstra (WD) method, which chooses paths based on current link utilization, using the GÉANT 2019 topology [115]. The evaluation testbed was built using Mininet [77], Open vSwitch [78], Ryu SDNC [59], Python-based GLBA, DLBA, and GSDNC, and Keras/TensorFlow [116] for the DRL algorithms implementation (extending the testbed used in [A1]).

HDRL-LB consistently improved throughput, aggregate data volume, and network availability compared to the baseline WD, especially under high load. For Transport Control Protocol (TCP) and User Datagram Protocol (UDP) traffic, throughput increased by $\sim$12% and $\sim$10%, and the number of missed flows was reduced by nearly 50% and 30%, respectively. These improvements came from better load balancing and reduced congestion. Moreover, the impact of GLBA and overlay policies cannot be neglected. The evaluation against an algorithm variant without global supervision showed substantial improvements in reward distribution across agents, translating into better aggregate performance when GLBA coordination is implemented.

3.2.4.1 3DQR

Principles Paper [A3] introduces the 3D QoS Routing (3DQR) framework, which extends the concepts proposed in [A2] towards QoS-aware routing in integrated 3D networks comprising Terrestrial, Aerial, and LEO Satellite network segments. As in the HDRL-LB case, 3DQR combines hierarchical multi-controller SDN with MADRL for domain-level and E2E network optimization (cf. Fig. 3.5). It aims to improve load distribution and flow acceptance rate, while accounting for flow QoS constraints, traffic priorities, and operator objectives. Moreover, to enable its application in mobile UP, the adopted model for the QoS requirements of user data flows aligns with the 5G QoS Identifier (5QI) classes proposed by 3GPP [117]. 3DQR also seeks to reduce broken paths and path reconfigurations caused by infrastructure mobility, thereby limiting session disruptions and the resulting SDN CP overhead.

To achieve these goals, 3DQR proposes following major innovations:

- overlay metrics to characterize overlay graphs (i.e., the topology perceived at the global network level) to enable E2E QoS-aware routing in the network;
- DRL-GNN Agent (DGA) – AI agents, implementing DRL algorithms [114] for intelligent routing decisions meeting the composite goals as described above. Each agent is embedded within centralized routers – Domain QoS-aware Router (DQR)/Main QoS-aware Router (M-QR) – that interact with SDNC to enforce routing decisions and observe their impact on the network;

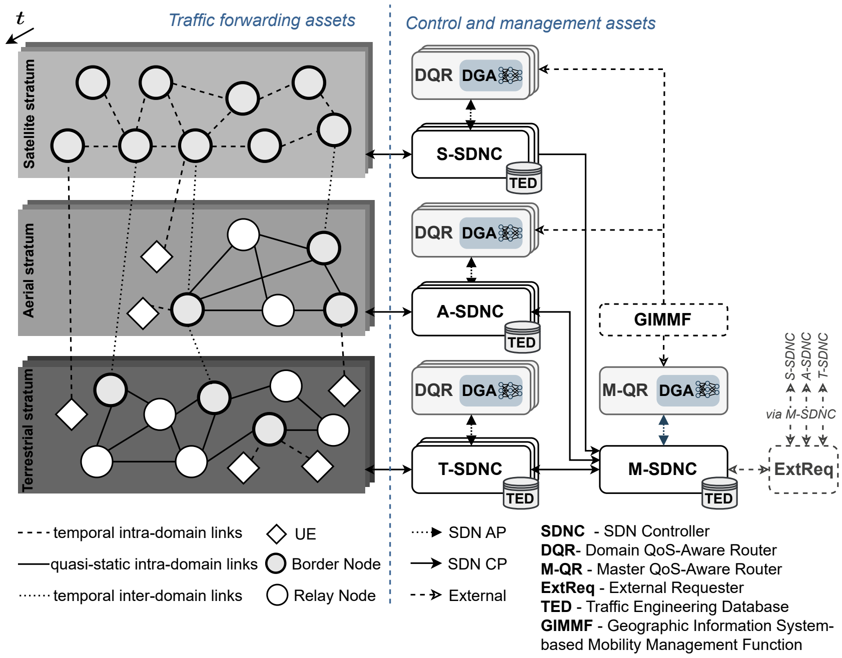


Figure 3.5: 3DQR entities and interactions (see [A3] for details)

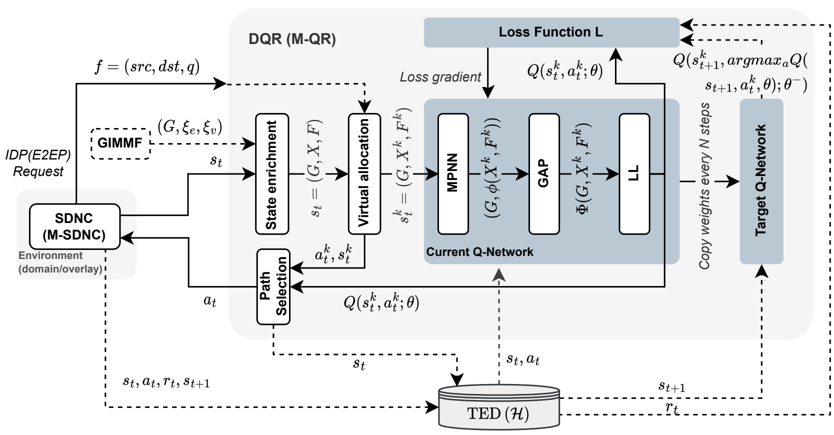


Figure 3.6: Internal agent structure and information flow (see [A3] for details)

- Graph Neural Network (GNN)-based agent architecture comprising Message Passing Neural Network (MPNN) [118] and Global Attention Pooling (GAP) [119], which enables reasoning based on both graph features and structure, while providing resilience to topology changes (cf. 3.6);
- inclusion of NTN topology predictions, based on satellite trajectories, to improve agent decision-making in time-variant environments;
- procedure for E2E QoS-aware path setup for user flows and complementary agents' training procedures to improve their QoS-awareness.

The proposed QoS-aware routing relies on three main mechanisms: Shortest-Path (SP) routing for candidate paths calculation, path evaluation by DRL agents to select a candidate path that contributes the most to the predefined objectives, and E2E path setup procedure including QoS requirements verification at domain and E2E levels. The algorithm details are described in Sections 4.2 and 5 of the paper [A3].

Evaluation & Results The 3DQR framework was evaluated using a proprietary event-driven simulator implemented in Python 3.10 with SimPy 4.1.1 [120], PyTorch 2.5.0, PyTorch Geometric 2.6.1 [121], Skyfield 1.48 [122], and public Starlink Two-Line Elements (TLE) orbit data [123] to derive satellite nodes trajectory and topology. Four evaluation dimensions were considered: complexity, performance, transferability of agents to previously unseen topologies, and the impact of topology-prediction accuracy. The experiments covered over 30 topologies comprising synthetic TN graphs and subsets of Starlink LEO constellations. The evaluated QoS classes based on 3GPP 5QI [117], that corresponded to key NTN services, namely, conversational voice, conversational video, buffered video streaming, and Aircraft to Everything (A2X) / aircraft telemetry messages.

In terms of complexity, 3DQR features higher complexity than the SP routing. The gap, however, decreases as the number of domains grows due to the distribution of operations across multiple SDNCs. For smaller networks, the complexity of 3DQR approaches that of C-SDN SP routing while delivering much better performance. Despite the overhead, 3DQR significantly outperformed baseline methods by reducing flow rejection by up to 13.5% in highly congested scenarios, improving load balancing by up to 52% in sparse topologies, and enabling effective QoS-aware traffic prioritization through coordinated intra-domain and inter-domain optimization. The agent's transferability tests showed consistently strong performance across most test scenarios. Comparable performance between native and transferred models confirmed the strong generalization capability of 3DQR across diverse topologies, particularly important for highly dynamic, time-varying network environments. The algorithm evaluation is wrapped with examination of the impact on topology prediction accuracy on 3DQR performance, showing substantial performance degradation for a rarely updated topology graph.

Finally, [A3] discusses practical challenges regarding 3DQR deployment related to: a) coordination overhead related to SDN CP distribution and control granularity [124], b) resource utilization and flow fairness due to fixed resource reservations; c) DRL cost-benefit tradeoff in sparse topologies; d) need for proactive flow rerouting to avoid SLA violations in random conditions.

3.3 Slice Admission Control (SAC) – summary of paper [A4]

3.3.1 Role in MANO and TE

5GS introduced Network Slicing (NS) paradigm, which requires specialized support from the Management and Orchestration (MANO) framework (cf. Section 4.4). One such mechanism is Slice Admission Control (SAC), which determines whether to accept or reject a network slice (i.e., Network Slice Instance (NSI)) 's deployment request based on factors such as required resources, priority, and Slice/Service Type (SST). The orchestration of admitted slices is then typically handled by the ETSI MANO framework. Since slices share a finite common resource pool, efficient resource allocation and Slice Admission Control (SAC) mechanisms are essential for SLA assurance, congestion control, fairness, and maximizing MNO profit [125]. In the SDN context, SAC acts as a complementary mechanism to TE, focusing on whether sufficient network capacity is available to deploy a slice, while TE optimizes the utilization of resources for admitted slices.

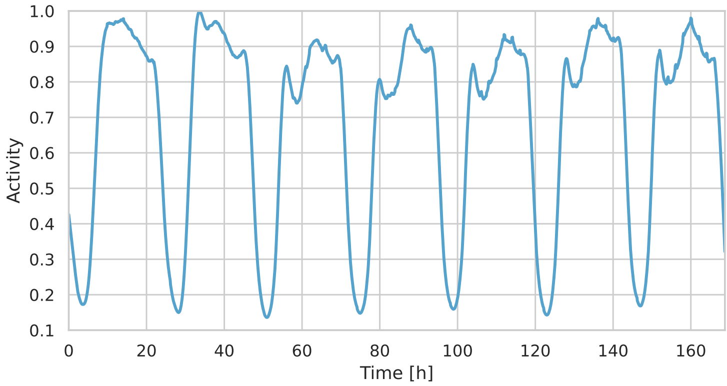


Figure 3.7: Normalized traffic intensity for a week (obtained from the production network)

NSI requests may include complex requirements related to capacity, temporal and spatial availability, security, application support, and many more. In practice, high-level requirements can be translated into low-level resource demands. The resource consumption of an NSI comprises a static factor related to the deployment footprint and a dynamic factor that depends on the activity of attached terminals. Regarding the latter, human-generated traffic exhibits strong Time-of-Day (ToD) dependence (cf. Fig. 3.7), with characteristic day-night and weekly variations, whereas traffic generated by Internet of Things (IoT), Industry 4.0, or monitoring devices is comparatively stable over time [126–128]. Exploiting ToD activity patterns can therefore improve SAC efficiency by incorporating expected resource consumption peaks into the decision-making process.

3.3.2 Time-of-Day-Aware Slice Admission Control (TASAC)

Principles Paper [A4] focuses on SAC strategies to improve infrastructure utilization. Recent studies have demonstrated the effectiveness of Reinforcement Learning (RL) and DRL approaches for SAC and resource allocation [129–134], which improved upon traditional greedy approaches in terms of profit, QoS assurance, and adaptability [114, 135, 136]. The SoA SAC solutions, however, typically assume static resource consumption and ignore temporal variations in human activity patterns and ToD, leading to resource over-provisioning.

This limitation is addressed in paper [A4], which proposes Time-of-Day-Aware Slice Admission Control (TASAC) algorithm, which incorporates information about requested slice resources, slice type, and current and predicted resource consumption derived from ToD activity models (cf. Fig. 3.8). By leveraging ToD-dependent traffic dynamics, TASAC aims to improve resource utilization and reduce SLA violations by enabling robust SAC decisions. For long-lived slices, ToD prediction helps mitigate congestion during peak hours and dynamically adjust allocated resources for short-lived slices (< 24 h) based on traffic trends. To derive an admission policy, TASAC uses the DDQN algorithm as the basis and extends it to incorporate the ToD user activity patterns. In the considered model, Enhanced Mobile Broadband (eMBB) slices are treated as ToD-dependent, while Massive Machine Type Communications (mMTC) slices are assumed to exhibit nearly constant demand. Moreover, as the resources needed for NSI deployment are typically split into computing, storage, and connectivity (which scale differently with activity), TASAC treats resources in a generic manner to enable flexible adaptation.

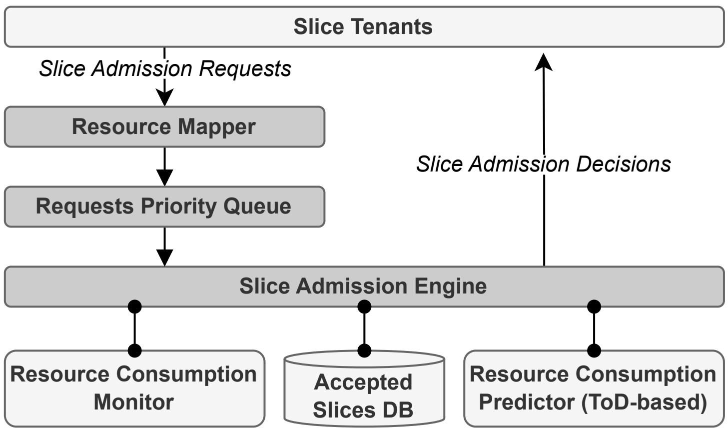


Figure 3.8: High-level view of TASAC architecture described in [A4]

Evaluation & Results TASAC evaluation focuses on connectivity resource type, i.e., bandwidth. The algorithm has been evaluated using an in-house event-driven Python-based simulator based on the SimPy [120] and Keras [116] for DRL algorithm implementation. The TASAC's performance has been assessed relative to the traditional Deep Q-Network (DQN)-based SAC for various ratios of arriving eMBB and mMTC slice deployment requests and a fixed resource pool.

The results demonstrate that TASAC achieves higher rewards from the beginning of training, over 25% better average resource utilization, and converges around 100 episodes faster than the DQN baseline. When algorithms converge, TASAC admits slices requiring approximately 60% more resources on average than the baseline, suggesting substantial potential gains for MNOs. Moreover, the experiments under unbalanced eMBB and mMTC slices show consistent gains in terms of convergence and resource utilization, with the best performance observed in mMTC-dominated scenarios.

Overall, the results confirm that ToD-aware modeling enables more efficient long-term resource utilization across when slices with various traffic properties are concerned. This observation is particularly important in the context of dynamically scheduled slices that can be postponed (e.g., database migrations, aggregated monitoring data upload/download), as it enables

MNOs to improve resource utilization and increase monetary gains. In the context of SDN and TE ([A2], [A3]), TASAC can operate as a mechanism to admit slices (groups of users generating UP traffic) based on the total capacity of the domain or E2E network. The capacity, corresponding to TASAC's resource pool, can be obtained, e.g., by using max-flow estimation methods.

3.4 Summary of research contributions

The key contributions of the thesis author presented in paper **[A2]** are as follows:

- Design HDRL-LB TE framework, for load balancing and throughput maximization in hierarchical multi-controller SDN environments. The architecture introduces a two-level control system combining hierarchical Multi-Agent Deep Reinforcement Learning (MADRL) setup to concurrently optimize the network at both domain- and global levels;
- Design and introduction of topology, metrics, and CP operations abstractions (i.e., overlay graphs and metrics) for scalable operations, reduced network overhead, and support for multi-provider networks with limited domain visibility and/or control;
- Enhanced DDPG-based routing with DDQN critic and time-aware state representation to improve convergence and adaptation to dynamically changing traffic patterns;
- Implementation of the HDRL-LB framework using open-source emulation tools (i.a., Mininet, OVS, Ryu, Tensorflow);
- Design of test scenarios, execution, and evaluation of HDRL-LB, demonstrating significant improvements in throughput, network carrying capacity, and load balancing compared to weighted Dijkstra routing in realistic SDN emulation environments;
- Validation of the influence of hierarchical DRL coordination, demonstrating significant improvements of reward distribution across domain agents, reduces congestion, and decreases missed flows in high-load scenarios, improving the overall performance and network stability.

The key contributions of the thesis author presented in paper **[A3]** are as follows:

- Design of 3DQR, a novel TE framework for future 3D networks that integrates H-SDN, hierarchicalMADRL, Graph Neural Network (GNN), and network topology prediction to support QoS-aware end-to-end routing and path allocation across terrestrial, aerial, and satellite segments. To the authors' knowledge, 3DQR was the first integration of MADRL and GNNs for centralized routing and path allocation in SDN-based 3D mobile networks;
- Proposal of new overlay metrics (metrics abstraction) for scalable E2E QoS-aware operations in H-SDN;

- Design of a hierarchical DRL-GNN Agent (DGA) routing and path allocation strategy in which domain and global agents use message passing, attention-based graph representations, and topology-persistence/expiry information to reason over dynamic, variable-size 3D network topologies;
- Formulation of QoS-aware routing and path allocation problem (time-expanded minimum-cost multi-commodity flow) in 3D mobile networks;
- Design of a MADRL algorithm that jointly considers flow acceptance, load balancing, QoS-class prioritization, and rerouting-cost minimization, while optimizing load distribution in the network. The mechanism includes a modular E2E routing workflow combining global overlay-path computation with intra-domain path allocation and verification, enabling coordinated routing across heterogeneous TN and NTN domains;
- Implementation of 3DQR framework comprising event-driven H-SDN simulator, NTN topology generator based on real-life satellite trajectory information
- Design and execution of comprehensive test scenarios of 3DQR for various topologies comprising generated TN and Starlink NTN constellations.
- Demonstration of performance gains over SoA baselines, achieving approximately 13% reduction in flow rejection rate and up to 50% improvement in traffic distribution and load balancing. The evaluation further shows effective QoS-class prioritization for high-priority traffic, alongside reduced path reconfiguration frequency and lower SDN control-plane overhead;
- Validation of generalization and transfer capability of the 3DQR agents, demonstrating consistent performance gains when models trained on one environment are applied to previously unseen network topologies and conditions.

The key contributions of the thesis author presented in paper **[A4]** are as follows:

- Design and development of a novel Slice Admission Control (SAC) algorithm – Time-of-Day-Aware Slice Admission Control (TASAC) based on Double Deep Q-Network (DDQN) for 5G network slicing environments.
- Introduction of Time-of-Day (ToD)-dependent resource consumption awareness into the SAC process, enabling more accurate admission decisions for slices with traffic patterns dynamically changing according to day-night cycles.
- Design and implementation of a simulation framework for evaluating the TASAC algorithm under different traffic and slice-type scenarios.

- Demonstration, through simulation-based evaluation, that the proposed TASAC approach significantly improves network resource utilization and slice admission efficiency compared to baseline methods, while limiting SLA violations.

Chapter 4

SDN and SDN-TE applications in 6G Networks

This section presents the author's research contributions related to 6G network architectures based on proposed SDN and SDN-TE. These include complementary works on emerging 6G use cases ([A5]), proposals for mobile User Plane (UP) and Control Plane (CP) architectures, as well as performance evaluation of the most popular standardization-compliant MANO solution.

4.1 6G Use Cases– summary of paper [A5]

4.1.1 Landscape

One of the key objectives of the 5GS was to open the mobile network ecosystem to vertical industries, enabling deep integration with the network and supporting specialized services built on mobile network capabilities. This design decision was primarily driven by the opportunity to create new business models and revenue streams for mobile network operators. Since the introduction of 5GS, 3GPP has defined six service classes, including the foundational eMBB, Ultra-Reliable Low-Latency Communication (URLLC), and Massive Internet of Things (MIoT), as well as Vehicle to Everything (V2X), High-Performance Machine-Type Communications (HMTC), High Data rate and Low Latency Communications (HDLLC), and Guaranteed Bit Rate Streaming Service (GBRSS), introduced during subsequent standardization stages [11]. These support a broad range of vertical applications across economic sectors such as entertainment (Augmented Reality (AR)/Virtual Reality (VR), high-definition video streaming), Industry 4.0 (robotics and factory automation), civil engineering (smart buildings and smart cities), agriculture (precision farming and photogrammetry), logistics (cargo delivery and autonomous transport via vehicles and drones), and many more. It is evident that 6GS will further extend the 5GS paradigm by expanding service support, simplifying integration, enhancing performance, and enabling more flexible service customization [12]. Collectively, these capabilities are expected to support advanced use cases grouped into six categories: Immersive Communication, Massive Communication, Hyper Reliable & Low-Latency Communication, Ubiquitous Connectivity, AI & Communication, and Integrated Sensing & Communication. Within these categories, the 3GPP has already identified over 200 specific use cases, each imposing distinct requirements on future network mechanisms and architectures [12].

4.1.2 Advanced Air Mobility (AAM)

One service type that has not yet achieved widespread commercialization in 5G, despite its significant market potential [137], is AAM-based services. AAM is a class of services using both manned and pilot-supervised/autonomous unmanned aircraft of various sizes [138] for human mobility, goods mobility, data collection, and task/process automation (cf. [A5] for example applications within each group). Due to operation in air in rural, suburban, and urban spaces, AAM faces multiple requirements related to safety, security, resilience, regulation, scalability, flexibility, and environmental impact [139].

First, AAM will be deployed within the Unmanned Aircraft Systems Traffic Management (UTM) ecosystem [139], which comprises services supporting aerial missions and is typically operated by or coordinated with aviation authorities. Second, airspace operations are subject to strict regulatory constraints, which impose additional functional requirements, including strong identity management, authentication and authorization, reliable tracking and location integrity, support for safety-critical control and monitoring, UAV–UTM information exchange, dynamic airspace management, and deeper integration of the communication network into UTM processes beyond mere data transport. In addition, AAM services require high reliability, performance, and differentiated QoS.

4.1.2.1 3GPP standardization efforts and research projects

The paper [A5] presents in detail the standardization efforts conducted within 3GPP Releases 17-19 that build the foundation for AAM services in 5GS. Stage 1 studies [140–143] identified 23 Unmanned Aerial Vehicle (UAV) use cases and corresponding requirements related to QoS, UAV video services, autonomy, swarm operations, Command and Control (C2) communication, and resource management, pre-flight feasibility checks, geofencing, and real-time C2 and trajectory monitoring. Stage 2 studies addressed network- and application-level aspects of integration [144, 145], including identification, authentication, authorization, tracking, QoS provisioning, C2 control, UTM interaction, and edge/cloud deployment considerations. Key outcomes were standardized in [146, 147], introducing a joint Public Land Mobile Network (PLMN)–UTM architecture and the Unmanned Aviation System Network Function (UAS NF) as a mediation function. Further work [148, 149] extended this with inter-UAV communication, UAV–GCS sidelink, UAV identifier broadcasting, and enhanced detect-and-avoid and mobility support. Security aspects were also standardized (authentication, C2 security, and secure UAS–UTM interactions, inter-UE C2 security) [150–152]. Finally, Stage 3 specifications define UAS NF APIs and a UTM mediation module enabling joint PLMN–UTM authentication and authorization [153][154].

3GPP standardization in AAM and UAV services has also been driven by several EU and international projects. The Horizon 2020 5G!Drones [155] project validated 5G-enabled UAV use cases, KPIs, and joint 5GS–UTM architectures supporting roaming for cross-border operations [156][157]. PriMO-5G [158], ETHER [159], and SESAR GOF 2.0 [160] investigated UAV-enabled situational awareness, TN-NTN integration, advanced air traffic management, and telecom–UTM interoperability mechanisms [161]. The global efforts, including the Open Generation 5G Consortium [162] and GSMA–GUTMA ACJA [163], focused on testbeds and data exchange frameworks between mobile networks and UTM systems.

4.1.2.2 Provided support and identified gaps

Overall, 5GS provides a strong technological foundation for early AAM deployment, it is still insufficient for fully scalable, safety-critical, and highly autonomous aerial operations. Key identified limitations include insufficient support for applications requiring extremely high data rates and ultra-low E2E latency, challenges in seamless mobility and handover for high-speed 3D aerial movement, and incomplete integration between PLMN and UTM/aviation ecosystems. Additional gaps concern scalable support for dense UAV traffic and swarm operations, real-time situational awareness, integrated sensing and communication, and advanced AI-driven network automation.

Finally, the envisioned evolution of network capabilities in 6GS is presented in the context of AAM requirements [2, 5, 164, 165]. The key upgrades include improved performance (data rates, latency, jitter, reliability), positioning accuracy, mobility support (relative velocity up to 1000 km/s compared to 500 km/s in 5G), and a higher device density per km^3. 6GS might also address coverage deficiencies and data rate limitations in rural areas by integrating with NTN systems.

4.2 User Plane (UP) Architecture – summary of paper [A6]

Standardization and research bodies have already started studies on 6G architecture, requirements, and use cases [2, 12, 166, 167], which are expected to be consolidated by the end of H1 2027 [168]. There is already a strong consensus that the next generation of networks should provide improved performance, flexible, fine-grained, and resilient traffic control, and support extensions for specific vertical domains [10]. These requirements imply the evolution of User Plane (UP), i.e., the system part responsible for user traffic handling. To address these challenges, 6G is envisioned to incorporate intelligence by design (i.e., via native-AI) and implement a cloud-native paradigm. 6GS is expected to reuse and extend existing 5GS components and architectural paradigms, including softwarization, Service-Based Architecture (SBA), or Control and User Plane Separation (CUPS) [117].

4.2.1 UP architecture in 5GS & key issues

In 5GS, the UP functionality is mostly implemented by three Network Functions (NFs) (cf. Fig. 4.1):

- **User Plane Function (UPF)** – handling user data forwarding and processing
- **Session Management Function (SMF)** – managing session setup, modification, and termination by using Packet Forwarding Control Protocol (PFCP) protocol [169]
- **Policy and Charging Function (PCF)** – defining policy rules in terms of QoS, charging, access rules, etc. and supplementing them to Session Management Function (SMF) for enforcement

5G UPF can implement flexible, service-specific packet processing functions such as buffering, classification, routing, marking, monitoring, and QoS enforcement. The major limitation of the 5G approach is the use of the GPRS Tunneling Protocol (GTP), which introduces stateful overlay tunnels to carry user traffic between the Radio Access Network (RAN) nodes and the Data Network (DN) termination (4.2).

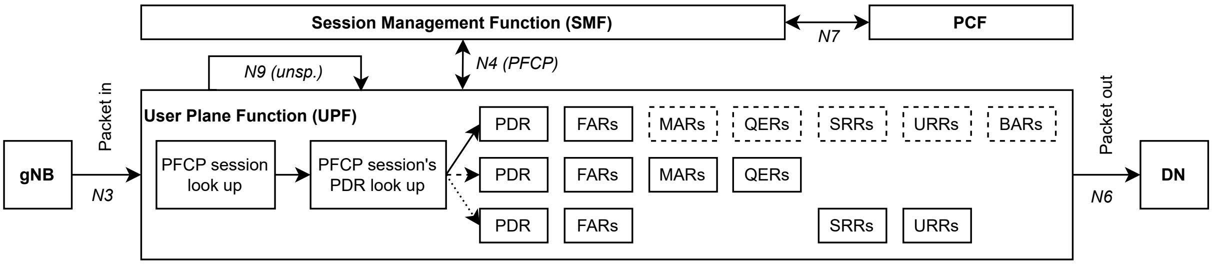


Figure 4.1: 5G UPF key components [117]

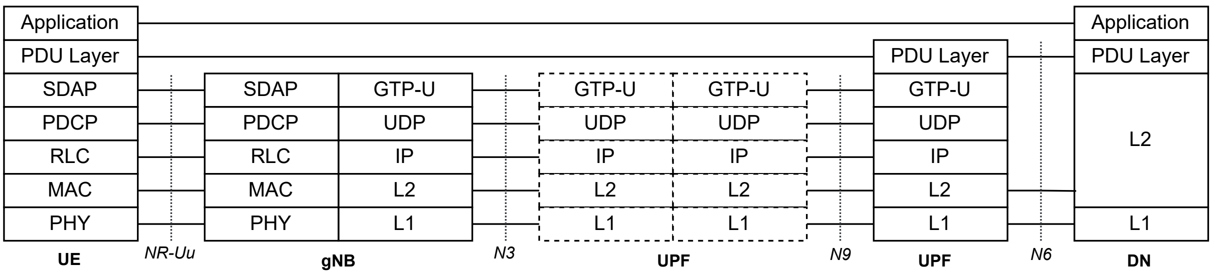


Figure 4.2: 5G UP protocol stack [117]

The approach suffers from several critical limitations, including signaling overhead due to tunnel maintenance during mobility and session changes (e.g., handover-induced tunnel updates or re-establishment), limited routing flexibility due to endpoint-based forwarding with constrained TE options, reliance on anchored mobility for session continuity, and limited Service Function Chaining (SFC) support, as it cannot natively encode chains of NFs for packet processing. These issues will further worsen as the transport network continues to disaggregate.

This issue has already been noticed by 3GPP [12], which considers merging the RAN nodes with UPFs to create Access Node User Plane (ANUP) to diminish signaling overhead regarding user session management existing in 5GS [13]. The second considered upgrade refers to improved programmability and the inclusion of the SRv6 protocol as an underlying protocol for GTP or as a GTP substitute [12]. While SRv6 provides many benefits, such as flexible Service Function Chaining (SFC) and TE, it will impose significant data overhead due to the encoding of network path information and operations within packet headers [14].

4.2.2 Evolution towards 6G and the role of SDN

In this context, SDN can bring substantial benefits for 6G UP design. First, it can be used to implement logically centralized control over SRv6 behavior in the network [15]. Second, it can be used to support GTP protocol [16]. Third, beyond conventional IP-based routing, SDN facilitates the implementation of advanced traffic steering mechanisms, including multi-path forwarding for improved resilience and mobility handling, fast path reconfiguration in highly dynamic environments (e.g., NTN scenarios), traffic multiplication and branching, mobility of two path ends, and fine-grained per-flow control. It also allows the incorporation of contextual information for routing decisions and exploitation of external network intelligence (e.g., cloud AI models). Furthermore, SDN enables the seamless integration of advanced TE functions, such as load balancing, congestion reduction, latency minimization, energy efficiency, and network-wide QoS/QoE optimization, thereby mitigating the inherent complexity and signaling overhead associated with legacy distributed TE protocols (e.g., RSVP-TE, SRv6).

While 5G UPF incorporates SDN-like principles such as centralized control, programmable forwarding, and CUPS, it is not fully aligned with SDN standards like OF [18]. Whereas OF can support some functions (e.g., flow matching and forwarding rules), it lacks native support for key QoS mechanisms such as delay measurement, buffering control, and full QoS enforcement, limiting its ability to fully implement 5QI enforcement. Moreover, in addition to SDN scalability and resilience issues (cf. Section 2), heterogeneous multi-domain environments pose interoperability challenges and make end-to-end QoS-aware TE difficult, especially in multi-stakeholder networks.

To address the 6G UP design challenges, several ideas have been proposed. These include functionally split UPs with distributed UPF components [170], "organic network" concepts that treat control procedures as modular processes [171], and architectures where SDNCs coordinate multiple UPFs or switches for end-to-end flow management [172]. Other approaches include OVS-based slicing using Virtual Local Area Networks (VLANs) [173], Programming Protocol-Independent Packet Processors (P4)-based programmable UPF systems such as Aether [174], or intelligent UP with In-Network Computing capabilities [175]. These approaches, however, lack clear integration of AI and support for integrated 3D networks, which are inherent to 6GS.

In parallel, to handle optimization complexity of UP in 6G, TE is increasingly being enhanced using AI/ML methods, particularly DRL agents [104, 176–178]. However, there is no unified framework that integrates SDN-based 6G UP with DRL-based TE while jointly addressing scalability, coordination, and end-to-end optimization challenges. Moreover, DRL, despite its strong potential to optimize UP in 6G networks, faces significant challenges in efficiency, robustness, transferability, observability, and generalization.

4.2.3 SDN-UP with DRL-TE (SUP-DTE)

concept To address the above-described concerns, paper [A6] introduces the SDN-UP with DRL-TE (SUP-DTE) framework compliant with 3GPP UP standardization, integrating SDN and supporting DRL-based TE in mobile UP spanning Terrestrial, Aerial, and Satellite network segments. The SDN-UP with DRL-TE (SUP-DTE) architecture is compliant with the ETHER MANO framework

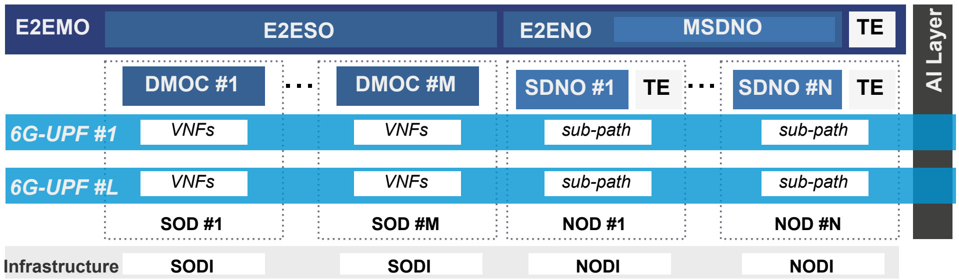


Figure 4.3: High-level view of SDN-UP with DRL-TE (SUP-DTE)[A6]

[179][180]. SUP-DTE comprises five layers responsible for: i) SDN-based network orchestration; ii) Service orchestration; iii) User Plane traffic control 6G User Plane Function (6G-UPF); iv) DRL-based TE; v) and AI model/data exchange and external stakeholders.

The architecture distinguishes Service Orchestration Domains (SODs) for Virtual Network Function (VNF) deployment and Network Orchestration Domains (NODs) for connectivity and routing across different administrative and technological domains. The system also introduces a proposal of 6G-UPF, in which the UP traffic is controlled by a hierarchical structure comprising a coordinator master UPF (mUPF) responsible for global session control and PFCP termination, and a chain of sub-UPFs (sUPFs) deployed across multiple NODs responsible for domain-level enforcement of steering and QoS. 6G-UPF also entails DRL-based TE agents operating at both hierarchy levels to optimize routing, QoS, and resource allocation. Moreover, SUP-DTE proposes a generic model for deployment of automated DRL-TE applications in hierarchical multi-domain SDN, following Monitor-Analyse-Plan-Execute based on Knowledge (MAPE-K) paradigm [181] and integrated with external AI models and data providers.

Overall, paper [A6] proposes a unified hierarchical SDN-UP architecture tightly integrated with DRL-based TE to address 6G requirements. As discussed in the paper [A6], the proposed design improves scalability, enables user-centric networking, supports autonomous optimization, and facilitates DRL-based optimization in multi-domain environments. Finally, the challenges concerning abstraction, coordination, and the complexity of real-world deployment are discussed, highlighting the need for further research before production deployment.

4.3 Control Plane Architecture – summary of paper [A7]

4.3.1 Core Network evolution directions

A key consensus is that the 6G core network should reuse existing 5G components and extend them as needed for continuous, incremental evolution that supports financial sustainability [10, 182]. Moreover, 5G assumptions, the Core Network (CN) only partially achieved characteristics of a cloud-native architecture. The key improvement areas indicated in the SoA include inter-NF communication efficiency [183], making NFs lightweight [184], and optimizing CP data flow [185].

Several architectural directions have already been proposed for next-generation mobile CN, e.g., user-centric networks deployed per-User Equipment (UE) [186], micro-service architectures decomposing NFs into µNFs [184], procedural CN [187], or stateless data-centric architectures [185][188]. The latter introduces stateless only NFs that execute procedures by directly manipulating network and user data stored in shared network databases. This approach reduces NF complexity and interaction overhead [185, 189], enables centralized data collection (beneficial for AI), and simplifies system development. The frequent database access by NFs and NFs synchronization, however, leads to CP latency overheads [190]. Nonetheless, the proposals remain highly disruptive, which conflicts with the goal of cost-effective evolution.

4.3.2 Hybrid SDN-based Data-Centric Core Network (HSDCN)

Paper [A7] introduces Hybrid SDN-based Data-Centric Core Network (HSDCN) architecture, which extends legacy 3GPP 5G CN with SDN mechanisms to enable incremental evolution and improve cloud-native capabilities. To achieve these goals, HSDCN introduces content-based and multi-path routing into the CN, implemented jointly by SDN and proxy-like extensions to NFs. The architecture comprises three interacting parts (cf. Fig. 4.4):

- **Hierarchical multi-controller SDN** – performing CP optimization (e.g., via Traffic Engineering), and configuration and implementation of content-based and multi-path routing;
- **Data-centric-enabled CN** – comprising legacy NFs and AI-native Intelligent NFs (INFs); The NFs can be optionally extended with Content-Based Routing Proxy (CBRP), which supports content-based forwarding functionality; The implementation details of content-based and multi-path routing are described in detail in Section III.A. [A7]
- **Intelligence Plane (IPL)** – a vertical plane comprising various AI-native functions, centralized data collection and management, and exposure of both data and capabilities to external systems (e.g., mobile CN).

While maintaining scalability and backward compatibility, HSDCN supports:

- **dataflow-driven design** instead of host-driven CN by decoupling NF communication from IP addressing
- **improving CP efficiency** via programmable signalling traffic steering and SDN-TE
- **dynamic SFC** in which, procedures defined as flows of content instead NF chains (dynamic and programmable signaling paths)
- **control messages replication** and routing to multiple NF branches – e.g., to facilitate parallel execution of procedures

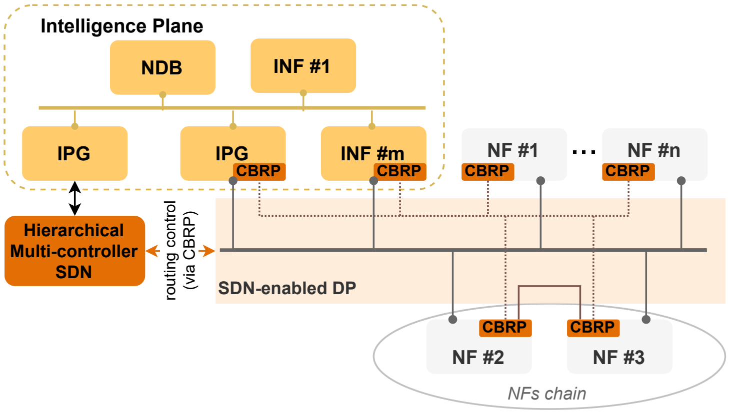


Figure 4.4: High-level concept of HSDCN [A7]

- **data-centric behavior**, allowing traffic to be handled based on content attributes rather than traditional addressing. This enables stateless, content-driven interactions between NFs and supports flexible, end-to-end service composition through programmable control abstractions.

The concept has been evaluated in a simplified simulation scenario focusing on procedure execution performance. MNOs most commonly deploy 5G architecture using Service Communication Proxy (SCP) as the mediator for NF interactions (star topology) for increased scalability [185]. In contrast, HSDCN enables direct NF-to-NF communication, similar to mesh-based deployments, while preserving scalability through its hierarchical control and routing mechanisms. The evaluation is based on 2,500 randomly generated fat-tree-like topologies (20–50 nodes) representing cloud, regional, and edge deployments spanning the distance of 50–250 km. The scenario includes NF chains of 5–20 functions with optimized NF placement.

Results show that HSDCN reduces end-to-end latency by approximately 26% and hop count by 19%, improving bandwidth efficiency. The benefits increase with a larger geographical scale due to higher propagation delays in SCP-based routing. HSDCN also exhibits lower latency and path-length variability, making it more suitable for time-sensitive scenarios such as edge and satellite networks while avoiding bottlenecks at SCP nodes. However, these gains come at the cost of increased complexity in content-based routing configuration and a higher deployment footprint of the proposed framework.

4.4 Management and Orchestration (MANO) – summary of paper [A8]

4.4.1 Evolution of MANO

Network Function Virtualisation (NFV) introduces a fundamental shift in telecommunications networks. In legacy systems, NFs were tightly coupled with underlying hardware. Network Function Virtualisation (NFV) decoupled software implementation of NF from the computation, storage, and networking resources and provisions them through a virtualization layer. While adding new capabilities to communications networks, NFV required a new set of management and orchestration functions to be added to the legacy model of operations, administration, maintenance, and provisioning. Consequently, MANO frameworks became essential to support these capabilities, with the European Telecommunications Standards Institute (ETSI) Network Function Virtualisation Management and Orchestration (NFV-MANO) architecture emerging as the most widely adopted framework, including 5GS.

In addition to its software-based nature, 5GS has further transformed the MANO in the telecommunications ecosystem due to the introduction of Network Slicing (NS) – technology, which enables multiple isolated virtual networks to run in parallel on shared infrastructure, where each network is tailored to specific service requirements [191]. To this end, 3GPP extended the Operations/Business Support System (OSS/BSS) part of the ETSI NFV-MANO architecture [192] by the components for orchestration and runtime management of 5G network slices. This significantly increases the load on MANO systems, which must handle large numbers of slices with diverse lifecycle and runtime constraints.

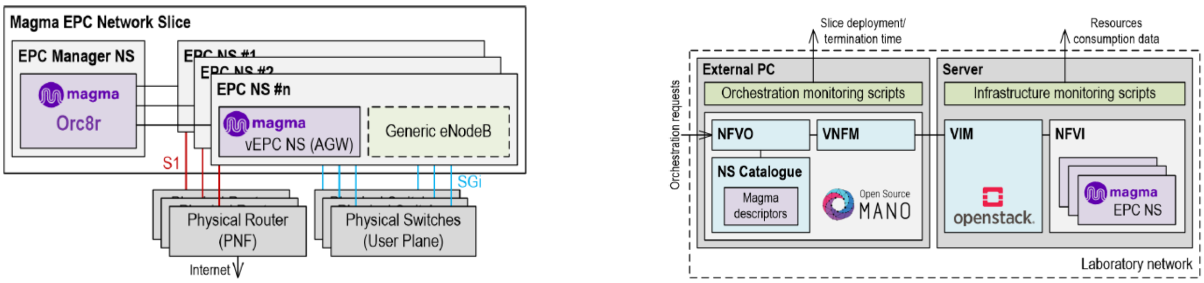
In this context, orchestration performance is critical, especially during the slice deployment, modification, and termination phases. These metrics directly impact service agility and are particularly important for slices, where orchestration delays can degrade service quality and lead to inefficient resource usage. Moreover, system behavior can also vary under different orchestration loads, which makes scalability a key requirement for MNOs.

It must be emphasized that both 3GPP and ETSI focused primarily on Performance Management (PM) and management Key Performance Indicators (KPIs)/Key Quality Indicators (KQIs)[193, 194], neglecting the orchestration performance. Today, despite several advances in PM, orchestration KPIs remain only partially standardized in 3GPP networks. Existing research proposes lifecycle-based KPIs to evaluate instantiation and termination agility [195], but these remain dependent on multiple factors such as slice complexity, number of VNFs, and infrastructure performance. Overall, while Network Slicing (NS) enables flexible 5G service delivery, its effectiveness is limited by orchestration scalability and lifecycle efficiency, and full-scale production validation remains an open challenge, due to partial observability provided by existing standards.

4.4.2 Scalability of orchestration

Paper [A8] evaluates lifecycle performance of an ETSI NFV-MANO-compliant orchestrator – Open-Source MANO (OSM) platform using a realistic Network Slice Template (NST) of a virtualized 4th Generation (4G) Evolved Packet Core (EPC) environment [196] (cf. Fig. 4.5a). It focuses on lifecycle KPIs, such as Slice Deployment Time and Slice Termination Time, and their scalability metrics for deployment and termination [195]. These measure how efficiently multiple network slices can be deployed or terminated under different loads, highlighting orchestration scalability challenges in real NFV systems. The paper [A8] also extends the earlier KPI frameworks by KPI for assessment of scalability of slice termination processes [195].

KPI measurement is based on OSS/BSS monitoring of NFVO interactions, using timestamped logs to capture procedure start/end times [197]. Both CPU and RAM utilization are also tracked during orchestration. The evaluation testbed comprised OSM Release 9.1 [198], OpenStack-based virtualized infrastructure, and in-house monitoring tools to track orchestration processes and consumed. The tests comprised deploying and terminating up to 18 Magma AGW-based vEPC slices [196] (cf. Fig. 4.5b) in a batch or sequentially.



(a) Magma Virtualized EPC slice

(b) Measurement testbed

Figure 4.5: OSM orchestration performance evaluation

Results presented in [A8] show that simultaneous orchestration significantly outperforms sequential orchestration. Deployment scalability metric is consistently <1 with best efficiency for batch sizes of 5 slices, while termination exhibits weaker gains, which peak for very small batch sizes (2–3 slices). Resource profiling shows that deployment exhibits an initial orchestration overhead phase, followed by steady memory growth and CPU peaks during provisioning, while termination shows a reverse pattern with memory release and CPU spikes at completion. One-by-one orchestration leads to higher memory consumption and lower efficiency than bundled execution. Overall, bundling reduces orchestration time and improves resource efficiency.

It must be noted that Magma-based slices are more complex than lightweight VM-based slices, leading to different scalability behavior [199]. Moreover, orchestration performance is highly dependent on slice complexity, infrastructure constraints, and deployment strategy. To this end, the full scalability assessment of MANO systems operating in realistic 5G/6G deployments remains an open challenge, especially under mixed virtualization stacks and stress conditions.

4.5 Summary of the research contributions

The key contributions of the thesis author presented in the paper **[A5]** are as follows:

- Comprehensive analysis and synthesis of knowledge concerning the AAM ecosystem, including stakeholders, missions, use cases, and communication requirements for future AAM services.

- Systematic review of ongoing 3GPP standardization activities related to AAM support in 5GS, including architectural, functional, and QoS aspects.

- Identification and synthesis of knowledge concerning 5GS support for AAM services, and exisiting 5G limitations.
- Analysis and synthesis of knowledge concerning 6G capabilities and technologies relevant to AAM, together with an assessment of their potential to address the communication and operational requirements of carrier-grade AAM services.

The key contributions of the thesis author presented in Article [A6] are as follows:

- Review and Synthesis of the SoA concerning key enablers and components for 6G User Plane (UP): SDN, 3GPP NFs and DRL-based TE methods with respect to 5G architecture features, future challenges, open issues and integration;
- Design of a novel 6G-enabled SDN-based User Plane architecture (SUP-DTE) integrating SDN, 6G User Plane Function (6G-UPF), and DRL-based TE within a hierarchical and distributed orchestration framework for scalable management of multi-domain integrated TN-NTN networks;
- Design of a 3GPP-compliant user-centric 6G-UPF concept based supporting fine-grained QoS-aware path management, AI-based TE, and rapid adaptation to mobility events;
- Design of a service-based, hierarchical DRL-enabled TE framework following Monitor-Analyse-Plan-Execute based on Knowledge (MAPE-K) approach, exposing services through a publish/subscribe communication model;
- Identification, analysis, and synthesis of key challenges associated with deploying DRL in carrier-grade SDN-based mobile networks, including scalability, observability, transferability, robustness, coordination, SLA maintenance, and generalization, followed by the proposal of mechanisms addressing these challenges;
- Analysis of the benefits, limitations, and open research challenges related to autonomous, intelligent, and scalable 6G User Plane operation in future mobile networks, including a review of challenges associated with the network-wide deployment and operation of the SUP-DTE architecture.

The key contributions of the thesis author presented in the paper [A7] are as follows:

- Comprehensive review of next-generation mobile core network architectures, requirements, emerging visions, and ongoing standardization efforts.
- Design of a Hybrid SDN-based Data-Centric Core Network (HSDCN) architecture, which combines 3GPP CN, a data-centric paradigm, Hierarchical multi-controller SDN (H-SDN) and content-based routing to support incremental and sustainable evolution of 5G towards 6G, preserving compatibility and exisiting NFs.

- Introduction of Intelligence Planes (IPLs) comprising INFs, generic AI services, Network Database (NDB) and its exposure towards core network and H-SDN orchestration.
- Design of a content-based and multi-path routing mechanism leveraging SDN for optimized handling of signaling traffic. The mechanisms enable traffic forwarding based on NF type, NF group, message semantics, or specific signaling content rather than IP addresses. This improves flexibility and enables implementation of advanced network mechanisms such as dynamic SFC, signaling branching, parallel procedure execution, and truly cloud-native routing decoupled from fixed host identities.
- Design and development of a simulator to evaluate the benefits of HSDCN architecture compared to the most popular SCP-based CN deployment.
- Quantitative concept evaluation over 2500 topologies, and various SFCs, indicating 26% reduction in E2E signaling latency, and an average 19% decrease of total path length (corresponding to bandwidth usage).

The key contributions of the thesis author presented in the paper **[A8]** are as follows:

- Deployment of OSM – ETSI-compliant MANO orchestrator and development of monitoring instrumentation for large-scale experimental validation of orchestration performance and scalability using resource-intensive MAGMA vEPC network slice templates;
- Extending orchestration performance KPIs by Slice Termination Time Scalability (STTS) metric, used to measure the scalability of network slice termination operations;
- Quantitative measurements of OSM performance and scalability regarding slice lifecycle operations, comprising bundled (simultaneous) and one-by-one deployment and termination of network slices;
- Scalability characterization showing that both deployment and termination times grow sub-linearly with increasing number of slices, which indicates good scalability properties of OSM. Network slice termination operations consume a surprisingly high amount of time, often exceeding 50% of the deployment time, highlighting an important practical orchestration challenge;
- Identification and analysis of practical limitations and future research directions, including in-depth investigation of orchestration bottlenecks, scalability under stress conditions, and performance evaluation across diverse infrastructure configurations and network slice templates.

Chapter 5

Discussion

The papers presented in Chapters 2-4 proposed several solutions centered around three research pillars crucial for next-generation mobile networks: **scalable Software-Defined Networking (SDN)**, and **scalable and intelligent SDN Traffic Engineering (SDN-TE)**, and their **applications in 6G network architectures**. The overarching goal of the presented research was to validate the dissertation thesis, which emphasizes three crucial aspects of modern mobile telecommunication networks, namely, **scalability, efficiency,** and **adaptability**:

Proposed Hierarchical multi-controller SDN (H-SDN) architecture and SDN-TE algorithms enable scalable, adaptive, and efficient traffic control suitable for next-generation mobile networks.

The SDN paradigm inherently provides network adaptability through network abstraction mechanisms and network-to-application adaptation. To address the scalability problem, it is proposed to use H-SDN architectural variant that distributes network control over multiple local SDN Controller (SDNC), which are managed by the higher-level centralized Hierarchical SDN Controller (HSDNC). Paper [A1] analyzes the benefits of adopting H-SDN compared to traditional centralized single-controller deployments, proving the substantial improvement of scalability at the cost of infrastructural resources. Moreover, increasing network distribution and control granularity, along with higher costs (due to additional controllers and higher capacity of network links), will **increase the coordination overhead** of control operations. For carrier-grade networks, it is essential to identify optimal trade-offs to prevent coordination overhead from outweighing the benefits of CP distribution, as described in Universal Scalability Law [124]. Evaluating such trade-offs, however, requires large-scale experimentation, which is difficult to perform in carrier-grade networks due to complexity and SLA constraints.

The TE algorithms described in papers [A4],[A2], and [A3] **improve network efficiency, performance, and MNO's profits** as compared to the SoA baseline methods. TASAC targets generic resource utilization improvements by considering natural user activity cycles into admission decisions, HDRL-LB improves traffic distribution to maximize total network throughput, and 3DQR jointly improves traffic distribution and flow acceptance rate while accounting for dynamically changing NTN topologies. **All algorithms exploit AI-based DRL algorithms** that enable adaptation to specific environmental characteristics for autonomous control. This feature is critical in heterogeneous 6G network deployments. The exploitation of hierarchical MADRL in [A2], and [A3], further contributes to this aspect. The hierarchical MADRL enables the implementation of agents that pursue individual, potentially conflicting goals across different network domains while adhering to the global optimization objectives imposed by the agent operating at the global level. The effectiveness of this approach is shown in both HDRL-LB and 3DQR

Furthermore, the SDN-TE approaches proposed in [A2] and [A3], **introduce abstractions** of:

- topology – the global-controller possesses a simplified network view, *overlay graph*, comprising domain gateways, inter-domain links, and virtual intra-domain links (i.e., abstracted connectivity between two gateways in one domain);
- network metrics – higher-level metrics calculated by local controllers based on telemetry data, and assigned to nodes and links of the *overlay graph*;
- SDN operations – inter-domain paths are obtained based on the *overlay graph*, but the internal routing is left to domain SDN controllers. Hence, inter-domain routing is reduced to gateway selection.

Using the above abstraction mechanism provides several benefits. First, the monitoring data volume exchanged between the domain and the global-level controllers is substantially reduced, increasing the scalability of SDN-TE. Despite certain information loss, the metrics abstractions enable efficient E2E network optimization (cf. [A2], [A3]). Second, the control distribution and the use of metrics and operations abstraction align well with the heterogeneous character of next-generation networks. In this context, domain SDN controllers can abstract local hardware and protocols, hiding domain-level complexity behind software interfaces exposed to upper network layers, which can be standardized. This feature, on the other hand, is essential to enable gradual network updates to SDN-based ones.

The **6G architectural solutions** proposed in papers [A6] and [A7] build on the proposed H-SDN architecture and can fully implement the introduced SDN-TE mechanisms. Both concepts account for domain-level heterogeneity, integration with external AI/ML systems, and adaptability by design. They leverage network data and AI/ML, in line with an AI-native network concept, which will play a fundamental role in improving model quality and further enhancing efficiency, autonomy, and adaptability. Paper [A7] further extends these capabilities by enabling low-cost customization of the mobile network core and by facilitating novel operational concepts such as parallel procedure execution, data-centricity, and dynamic SFC, which can be implemented via SDN-driven content-based and multi-path routing. These enhancements increase adaptability while maintaining compliance with the 3GPP system. Finally, driven by the increasing spatial availability of data centers at the edge, regional, and national levels, the importance of mobile UP disaggregation and control distribution has been recognized by Standards Developing Organizations (SDOs) [12, 13], which is well aligned with H-SDN principles.

Papers [A5] and [A8] focus more on exploratory aspects within the broader scope of the thesis. Although not explicitly emphasized in [A5], the use of SDN can provide several benefits for AAM implementations. Software-driven control facilitates the implementation of SDN applications dedicated to AAM missions support, such as mechanisms targeting seamless mobility (e.g., path setup prior to handover), context-aware routing and handovers (leveraging additional information about users, applications, networks, and environment), traffic branching and duplication (for improved resilience), or pre-allocation of network resources prior to mission execution (e.g., during the flight planning phase). Overall, the AAM highlights the complexity of 6G use cases and indicates that greater network flexibility will be essential to meet the diverse requirements of specific verticals. Paper [A8], on the other hand, evaluated a popular mobile network orchestration system – Open-Source MANO (OSM). Although not directly incorporated into subsequent studies, the evaluation highlights the importance of orchestration performance and scalability in modern mobile networks, which is also relevant to SDN-based systems.

The results and conclusions restated above, presented in more detail in papers [A1]–[A8], support the dissertation thesis and confirm its validity.

Despite evident gains, the proposed concepts introduce several limitations and open problems. First, the E2E performance of H-SDN is dependent on the performance of the domain SDNCs involved in the traffic control processes. To this end, in production deployments, the controllers will require relevant redundancy and resource headroom to maintain performance under variable conditions. This translates into infrastructure costs, which can become substantial in large-scale deployments, especially when link capacity requires scaling up. Moreover, the performance in heterogeneous networks, e.g., cooperation with non-SDN domains, might be less optimistic. While hierarchy and metrics abstraction partially solves this problem, the issue of E2E QoS enforcement becomes even more difficult to address.

Another aspect concerns AI-training costs. While papers [A2] and [A3] propose algorithms based on relatively simple AI model architectures, it must be emphasized that in operational networks, DRL models cannot be freely trained without risking SLA violations. Consequently, there is a strong need to deploy Network Digital Twin (NDT) solutions for model pretraining, resulting in significantly increased costs. This introduces a critical cost-performance trade-off that must be carefully evaluated when deploying AI-driven TE in networks generally.

Finally, considering the scale of the carrier-grade network, a single HSDNC and a single hierarchy level might not be sufficient to handle production workloads. While an additional level of hierarchy can be introduced above HSDNC, it would further increase system complexity, particularly in terms of QoS enforcement for network users. This would require another level of metric abstraction, which would need to be carefully assessed to determine whether it provides sufficient value for TE processes. Moreover, alternative mixed deployments should be considered, such as flat multi-controller SDN at the global level.

In addition to addressing the open problems identified above, future work should also extend the proposed methods toward emerging classes of AI algorithms, such as GNN, generative AI, and Large Language Models (LLMs), for integration into SDN-TE. While LLMs are primarily used in SDN for intent-based (i.e., policy-driven) network management, their broader potential for traffic engineering remains largely unexplored. The application of GNN can also be extended to provide network-level predictions for SDN-TE to improve real-time decision-making as well as network evolution (e.g., via GNN-based network capacity planning). Finally, the proposed architectures would require alignment with the official 6GS architecture once it is standardized and reaches a high maturity level.

Chapter 6

Summary & Conclusions

This dissertation investigated two crucial aspects of the mobile system evolution towards 6G: the potential of exploitation of **SDN within the mobile networks**, and **AI-based TE algorithms** for intelligent, scalable, and efficient traffic optimization in the network DP.

The presented research papers advance the SoA in next-generation mobile networking through the design and validation of scalable, adaptive, and AI-driven architectures based on H-SDN and DRL-enabled TE. The proposed solutions demonstrate that combining distributed SDN control with intelligent optimization mechanisms can significantly improve the performance, scalability, and adaptability of future mobile environments, including integrated 3D terrestrial–non-terrestrial networks.

The conducted analytical, simulation-based, and experimental evaluations confirm the feasibility and effectiveness of H-SDN and AI-native approaches for next-generation mobile networks. Although these approaches alleviate scalability limitations, they also introduce new challenges related to system complexity, inter-domain coordination, QoS assurance, and operational costs, underscoring the need for further investigation before large-scale deployment. Nevertheless, as discussed in Chapter 5, the results presented in papers [A1]–[A8] validate the dissertation thesis and confirm the potential of using hierarchical SDN architectures to build future mobile communication systems.

Summary of the **key achievements of this dissertation and contributions to the SoA** are as follows:

[C1] Design of SDN CP scalability model based on Flatto–Hahn–Wright (FHW) queueing to represent Hierarchical multi-controller SDN (H-SDN) system, definition and formalization of key SDN scalability performance metrics (controller utilization and Time-To-Serve (TTS)), inclusion of CP operational overhead into scalability modelling, proposal of resource consumption model based on controller utilization for the SDN CP (covering compute, memory, disk I/O, and bandwidth utilization), proposal of metrics for CP scalability estimation, implementation of the emulation testbed and successful model validation.

The description of work is presented in Chapter 2 based on paper [A1].

[C2] Design, implementation, and evaluation of the HDRL-LB TE framework for load balancing and throughput optimization in H-SDN environments, using hierarchical MADRL implementing enhanced DDPG mechanisms and time-aware state representations, demonstrating significant improvements in throughput, congestion reduction, and load balancing compared to conventional SDN routing approaches.

The description of work is presented in Chapter 3, Section 3.2 based on paper [A2].

[C3] Design, implementation, and evaluation of the 3DQR TE framework for QoS-aware routing and path allocation in integrated 3D TN-NTN networks, combining hierarchical MADRL, GNNs, topology prediction, and H-SDN for scalable routing across dynamic terrestrial, aerial, and satellite domains.

The description of work is presented in Chapter 3, Section 3.2 based on paper [A3].

[C4] Proposal, application, and validation of network topology and metrics abstractions for global level QoS-aware TE in H-SDN environments.

The description of work is presented in Chapter 3, Section 3.2 based on papers [A2] and [A3].

[C5] Design, implementation, and evaluation of a TASAC algorithm based on a DDQN, which introduces Time-of-Day (ToD) dependent resource consumption model to optimize SAC decisions. The simulation validation demonstrates improved network resource utilization and slice admission efficiency compared to baseline methods, emphasizing the importance of including ToD cycles into optimization processes.

The description of work is presented in Chapter 3, Section 3.3 based on paper [A4].

[C6] Systematic SoA analysis and knowledge synthesis concerning future AAM communication ecosystems, including end-to-end requirements, stakeholders, and use cases, together with a review of relevant 3GPP 5GS standardization efforts, current limitations, and envisioned support in 6G mobile networks.

The description of work is presented in Chapter 4, Section 4.1 based on paper [A5].

[C7] Design of an HSDCN architecture integrating 3GPP core networks, a data-centric paradigm, and H-SDN with content-based routing to enable a gradual core network evolution towards 6G. The framework enables programmable signaling traffic steering based on NF type, semantics, and content, supports dynamic SFC, signaling branching, and parallel procedure execution. Simulation-driven validation demonstrates reduced end-to-end latency and signaling path length compared to popular core network deployments.

The description of work is presented in Chapter 4, Section 4.3 based on paper [A7].

[C8] Design of a 6G-enabled SDN-UP with DRL-TE (SUP-DTE) architecture, integrating H-SDN, 6G User Plane Function (6G-UPF), DRL-based TE for optimized user traffic handling in distributed TN–NTN networks. The design introduces a 3GPP-compliant, user-centric 6G-UPF concept that enables fine-grained QoS-aware path management, and a service-based hierarchical DRL-TE framework that follows a MAPE-K approach for automated optimization at different system levels.

The description of work is presented in Chapter 4, Section 4.2 based on paper [A6].

[C9] Evaluation of performance and scalability of orchestration operations of ETSI-compliant OSM orchestrator using realistic and resource-intensive network slice templates (Magma vEPC), introduction of Slice Termination Time Scalability (STTS) KPI, and demonstration of sub-linear scaling of deployment and termination operations of OSM orchestrator.

The description of work is presented in Chapter 4 Section 4.4 based on paper [A8].

The thesis was carried out as part of an **industrial PhD programme** conducted in collaboration with **Orange Innovation Poland**, the research laboratory of **Orange Polska S.A.**, and **Warsaw University of Technology**. The research presented in this dissertation was performed within international European Union (EU)-funded and as a part of internal research initiatives conducted within the **Orange** group:

- Paper [A8] – '5G-DRIVE: 5G HarmoniseD Research and TrIals for serVice Evolution between EU and China'. Funding: Horizon 2020 Research and Innovation programme, Grant Agreement No. 814956.
- Papers [A5],[A6],[A2],[A7], and [A3] – 'ETHER – sElf-evolving terrestrial/non-Terrestrial Hybrid nEtwoRks'. Funding: Horizon Europe, Smart Networks and Services Joint Undertaking (SNS JU), Grant Agreement No. 101096526.
- Papers [A1][A4] – 'Beyond 5G: Multi-domain SDN for 6G User Plane'. Funding: Orange Innovation (France)

Although the majority of the above projects focused on low- to mid-Technology Readiness Level (TRL) research and Proof of Concept (PoC) validation, the obtained results constitute an important step toward the industrial adoption of the proposed solutions. In particular, the dissertation demonstrates how scientific methods can be combined with practical engineering to address key challenges of next-generation mobile networks. At the same time, the transition from research prototypes to fully operational and scalable deployments remains an ongoing challenge that requires further development, validation, and industrial integration efforts.

Chapter 7

Academic Achievements

This section outlines the author's key academic achievements up to the submission of the dissertation in 2026. As of Tuesday 9th June, 2026, based on Google Scholar, WUT Repository, and the list of academic journals and peer-reviewed international conference materials published by Ministry of Science and Higher Education (MSHE)[200], the author has achieved the following during the period:

- h-index: 9
- Total CiteScore: 61,4
- Cumulative MNiSW score: 1210 (25 works)
- Citations: 376

7.1 Publications

7.1.1 Journals

[J1] Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Evaluating Scalability of Hierarchical Multi-controller SDN: A Holistic Approach". In: *Journal of Network and Systems Management* 34.4 (May 2026), p. 107. ISSN: 1573-7705. DOI: 10.1007/s10922-026-10087-w
IF: 3.9; **Ministerial points: 70**

[J2] Robert Kołakowski et al. "Hierarchical Traffic Engineering in 3D Networks Using QoS-Aware Graph-Based Deep Reinforcement Learning". In: *Electronics* 14.5 (Mar. 2025), p. 1045. ISSN: 2079-9292. DOI: 10.3390/electronics14051045
IF: 2.6; **Ministerial points: 100**

[J3] Luis Blanco et al. "AI-Driven Framework for Scalable Management of Network Slices". In: *IEEE Communications Magazine* 61.11 (Nov. 2023), pp. 216–222. ISSN: 1558-1896. DOI: 10.1109/MCOM.005.2300147
IF: 8.3; **Ministerial points: 200**

[J4] Lechosław Tomaszewski and Robert Kołakowski. “Advanced Air Mobility and Evolution of Mobile Networks”. In: *Drones* 7.9 (Sept. 2023), p. 556. ISSN: 2504-446X. DOI: `10.3390/drones7090556`. (Visited on 02/27/2025)
IF: 4.4; **Ministerial points: 40**

[J5] Lechosław Tomaszewski and Robert Kołakowski. “Mobile Services for Smart Agriculture and Forestry, Biodiversity Monitoring, and Water Management: Challenges for 5G/6G Networks”. In: *Telecom* 4.1 (Mar. 2023), pp. 67–99. ISSN: 2673-4001. DOI: `10.3390/telecom4010006`. (Visited on 02/27/2025)
IF: 2.1; **Ministerial points: 5**

[J6] Lechosław Tomaszewski et al. “Mobile Networks’ Support for Large-Scale UAV Services”. In: *Energies* 15.14 (Jan. 2022), p. 4974. ISSN: 1996-1073. DOI: `10.3390/en15144974`
IF: 3.0; **Ministerial points: 140**

[J7] Slawomir Kukliński et al. “6G-LEGO: A framework for 6G network slices”. In: *Journal of Communications and Networks* 23.6 (Dec. 2021), pp. 442–453. ISSN: 1976-5541. DOI: `10.23919/JCN.2021.000025`
IF: 3.6; **Ministerial points: 70**

7.1.2 Conference Proceedings

[C1] Sławomir Kukliński, Robert Kołakowski, and Bartłomiej Mastej. “Migration-driven Energy Consumption Optimisation atop Cloud Continuum”. In: *2026 IEEE 12th International Conference on Network Softwarization (NetSoft)*. Berlin, Germany, June 2026. DOI: `(inpress)`
IF: –; **Ministerial points: 20**

[C2] Robert Kołakowski, Thierry Lejkin, and Victorien Romain. “Hybrid SDN-Based Data-Centric Mobile Core Network with Content-Based Routing”. In: *2025 International Conference on 6G Networking (6GNet 2025)*. Paris, France, Dec. 2025, pp. 126–130. DOI: `10.1109/6GNet68413.2025.11314080`
IF: –; **Ministerial points: 20**

[C3] Sławomir Kukliński, Robert Kołakowski, and Bartłomiej Mastej. “Mlops as a Service for AI-Native 6G Networks”. In: *2025 IEEE 11th International Conference on Network Softwarization (NetSoft)*. June 2025, pp. 61–66. DOI: `10.1109/NetSoft64993.2025.11080538`

IF: –; **Ministerial points: 70**

[C4] Robert Kołakowski, Slawomir Kukliński, and Lechosław Tomaszewski. "Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Networks". In: *2024 IFIP Networking Conference (IFIP Networking)*. June 2024, pp. 607–612. DOI: 10.23919/IFIPNetworking62109.2024.10619899
IF: –; **Ministerial points: 70**

[C5] Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "On 6G-Enabled SDN-Based Mobile Network User Plane with DRL-Based Traffic Engineering". In: *Artificial Intelligence Applications and Innovations. AIAI 2024 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2024, pp. 124–138. ISBN: 978-3-031-63227-3. DOI: 10.1007/978-3-031-63227-3_9
IF: –; **Ministerial points: 20**

[C6] Lechosław Tomaszewski and Robert Kołakowski. "On the Efficient Architecture for 6G System". In: *Artificial Intelligence Applications and Innovations. AIAI 2024 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2024, pp. 139–153. ISBN: 978-3-031-63227-3. DOI: 10.1007/978-3-031-63227-3_10
IF: –; **Ministerial points: 20**

[C7] Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Time-of-Day-Aware Slice Admission Control". In: *2023 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. Sept. 2023, pp. 199–204. DOI: 10.1109/MeditCom58224.2023.10266645
IF: –; **Ministerial points: 20**

[C8] Lechosław Tomaszewski and Robert Kołakowski. "Network Slicing vs. Network Neutrality – Is Consent Possible?" In: *Artificial Intelligence Applications and Innovations. AIAI 2023 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2023, pp. 77–90. ISBN: 978-3-031-34171-7. DOI: 10.1007/978-3-031-34171-7_6
IF: –; **Ministerial points: 20**

[C9] Lechosław Tomaszewski et al. "ETHER: Energy- and Cost-Efficient Framework for Seamless Connectivity over the Integrated Terrestrial and Non-terrestrial 6G Networks". In: *Ar-*

tificial Intelligence Applications and Innovations. AIAI 2023 IFIP WG 12.5 International Workshops. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2023, pp. 32–44. ISBN: 978-3-031-34171-7. DOI: 10.1007/978-3-031-34171-7_2
IF: –; **Ministerial points: 20**

[C10] Lechosław Tomaszewski, Robert Kołakowski, and Mirosław Zagórda. "Application of Mobile Networks (5G and Beyond) in Precision Agriculture". In: *Artificial Intelligence Applications and Innovations. AIAI 2022 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer International Publishing, 2022, pp. 71–86. ISBN: 978-3-031-08341-9. DOI: 10.1007/978-3-031-08341-9_7
IF: –; **Ministerial points: 20**

[C11] Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "Performance evaluation of the OSM orchestrator". In: *2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. Nov. 2021, pp. 15–20. DOI: 10.1109/NFV-SDN53031.2021.9665052
IF: –; **Ministerial points: 20**

[C12] Sławomir Kukliński et al. "A Novel Architectural Approach for the Provision of Scalable and Automated Network Slice Management, in 5G and Beyond". In: *Artificial Intelligence Applications and Innovations. AIAI 2021 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis, John Macintyre, and Lazaros Iliadis. Cham: Springer International Publishing, 2021, pp. 39–51. ISBN: 978-3-030-79157-5. DOI: 10.1007/978-3-030-79157-5_4
IF: –; **Ministerial points: 20**

[C13] Lechosław Tomaszewski et al. "High Mobility 5G Services for Vertical Industries – Network Operator's View". In: *Artificial Intelligence Applications and Innovations. AIAI 2021 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis, John Macintyre, and Lazaros Iliadis. Cham: Springer International Publishing, 2021, pp. 71–84. ISBN: 978-3-030-79157-5. DOI: 10.1007/978-3-030-79157-5_7
IF: –; **Ministerial points: 20**

[C14] Lechosław Tomaszewski, Sławomir Kukliński, and Robert Kołakowski. "A New Approach to 5G and MEC Integration". In: *Artificial Intelligence Applications and Innovations. AIAI 2020 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis, Lazaros Iliadis, and Elias Pimenidis. Cham: Springer International Publishing, 2020, pp. 15–24. ISBN: 978-3-030-49190-1. DOI: 10.1007/978-3-030-49190-1_2

IF: –; **Ministerial points: 20**

[C15] Lechosław Tomaszewski, Robert Kołakowski, and Paweł Korzec. "On 5G Support of Cross-Border UAV Operations". In: *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. June 2020, pp. 1–6. DOI: `10.1109/ICCWorkshops49005.2020.9145262`. (Visited on 02/27/2025)

IF: –; **Ministerial points: 70**

[C16] Lechosław Tomaszewski, Robert Kołakowski, and Sławomir Kukliński. "Integration of U-space and 5GS for UAV services". In: *2020 IFIP Networking Conference (Networking)*. June 2020, pp. 767–772. (Visited on 02/27/2025)

IF: –; **Ministerial points: 140**

[C17] Sławomir Kukliński, Lechosław Tomaszewski, and Robert Kołakowski. "On O-RAN, MEC, SON and Network Slicing integration". In: *2020 IEEE Globecom Workshops (GC Wkshps*. Dec. 2020, pp. 1–6. DOI: `10.1109/GCWkshps50303.2020.9367527`

IF: –; **Ministerial points: 20**

[C18] Sławomir Kukliński et al. "5G-UASP: 5G-based multi-provider UAV platform architecture". In: *2020 6th IEEE Conference on Network Softwarization (NetSoft)*. June 2020, pp. 242–246. DOI: `10.1109/NetSoft48620.2020.9165458`

IF: –; **Ministerial points: 20**

7.1.3 Whitepapers

[W1] Smart Networks and Services Joint Undertaking (SNS JU) 6G Architecture Working Group, "Towards 6G Architecture: Key Concepts, Challenges, and Building Blocks", DOI:10.5281/zenodo.15001377

7.2 Conference speeches

- *Integration of U-space and 5GS for UAV services.* 2020 IFIP Networking Conference, Paris, France, Jun. 2020
- *Performance evaluation of the OSM orchestrator.* 2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Virtual, Nov. 2021.
- *Network Slicing vs. Network Neutrality – Is Consent Possible?.* 19th Artificial Intelligence Applications and Innovations, Leon, Spain, 2023.

- *ETHER: Energy- and Cost-Efficient Framework for Seamless Connectivity over the Integrated Terrestrial and Non-terrestrial 6G Networks.* 19th Artificial Intelligence Applications and Innovations, Leon, Spain, 2023.

- *Time-of-Day-Aware Slice Admission Control.* Third IEEE International Mediterranean Conference on Communications and Networking (IEEE MEDITCOM 2023), Dubrovnik, Croatia, Sep. 2023.

- *Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Network.* International Federation for Information Processing Networking 2024 Conference (IFIP NETWORKING 2024), Thessaloniki, Greece, Jun. 2024.

- *On 6G-Enabled SDN-Based Mobile Network User Plane with DRL-Based Traffic Engineering.* Artificial Intelligence Applications and Innovations (AIAI), Corfu, Greece, Jun. 2024.

- *Hybrid SDN-Based Data-Centric Mobile Core Network with Content-Based Routing*, 2025 International Conference on 6G Networking (6GNet 2025), Paris, France, Dec. 2025.

7.3 Participation in Research Projects

- *5G-DRIVE: 5G HarmoniseD Research and TrIals for serVice Evolution between EU and China*, Role: R&D Scientist ; Funding: European Union, Horizon 2020 (2018-2021)

- *5G!Drones: Unmanned Aerial Vehicle Vertical Applications' Trials Leveraging Advanced 5G Facilities*, Role: R&D Scientist; Funding: European Union, Horizon 2020 (2019-2023)

- *MonB5G*, Role: R&D Scientist; Funding: European Union, Horizon 2020, (2019-2023)

- *ETHER: sElf-evolving terrestrial/non-Terrestrial Hybrid nEtwoRks*, Role: R&D Scientist; Funding: Horizon Europe Smart Networks and Services Joint Undertaking

- *ELIXIRION: rEaLIzing healthcare 4.0 eXploIting the 6G netwoRk evolutION*, Role: Scientific/Technical Consultant; Funding: Horizon Europe TMA MSCA Doctoral Networks

- *Beyond 5G: Multi-domain SDN for 6G User Plane* – Role: Project Manager, R&D Scientist; Funding: Orange France (2024-2026)

- *6G-Cloud: Service-oriented 6G Network Architecture for Distributed, Intelligent, and Sustainable Cloud-native Communication Systems*, Role: R&D Scientist, DevOps ; Funding: European Union, 6G Smart Networks and Services Joint Undertaking (SNS JU)

7.4 Obtained Funding & Grants:

- *ETHER: sElf-evolving terrestrial/non-Terrestrial Hybrid nEtwoRks*; Horizon Europe Smart Networks and Services Joint Undertaking (2023-2026)

- *ELIXIRION: rEaLIzing healthcare 4.0 eXploIting the 6G netwoRk evolutION*, Horizon Europe TMA MSCA Doctoral Networks (2023-2027)
- *Beyond 5G: Multi-domain SDN for 6G User Plane* – Internal Funding by Orange Polska (2024-2026)

7.5 Other Achievements

7.5.1 Contributions to telecommunication standards

[S1] ITU-T: Y.IMT2020-IBNMO: "Intent-based network management and orchestration for network slicing in IMT-2020 networks and beyond",

[S2] ITU-T: Y.IMT-2020-DL-AINW-fra: "A communication model for AI-based management in IMT-2020 and beyond",

[S3] ITU-T: Y.SLOA-arch: "End-to-end service level objective assurance architecture for future networks including IMT-2020",

[S4] ITU-T: Y.NSL-fra: "Framework for classifying network slice level in future networks including IMT-2020".

[S5] ITU-T: Y.Y.FMSC-SF: "Fixed, mobile and satellite convergence - Store and forward for IMT-2020 and beyond"

7.5.2 Research Deliverables

[D1] 5G-DRIVE: "D3.3: Final Report of eMBB Trials"

[D2] 5G-DRIVE: "D4.4: Final Report on V2X Trials"

[D3] 5G-DRIVE: "D5.2: Final report of 5G technology and service innovations"

[D4] 5G!Drones: "D1.3: System Architecture Initial Design"

[D5] 5G!Drones: "D1.4: Report on UAV business and regulatory ecosystem and the role of 5G"

[D6] 5G!Drones: "D1.6: 5G!Drones system architecture refined design"

[D7] 5G!Drones: "D1.6: 5G!Drones system architecture refined design"

[D8] 5G!Drones: "D3.1: Report on infrastructure-level enablers for 5G!Drones

[D9] 5G!Drones: "D3.3: 5G!Drones Enablers Software Suite"

[D10] 5G!Drones: "D5.6 Report on activities related to commercial exploitation and partnership development"

[D11] MonB5G: "D2.1: 1st release of the MonB5G zero touch slice management and orchestration architecture"

[D12] MonB5G: "D2.2: Techno-economic analysis of the beyond 5G environment, use case requirements and KPIs"

[D13] MonB5G: "D2.4: Final release of the MonB5G architecture (including security)"

[D14] MonB5G: "D3.3: Final Report on Platform Integration for the MonB5G AE/MS"

[D15] MonB5G: "D4.1: Initial Report on AI-Driven Techniques for the MonB5G Decision Engine"

[D16] MonB5G: "D4.2: Final Report on AI-driven Techniques for the MonB5G Decision Engine"

[D17] MonB5G: "D4.3: Report on Integration and testing of the MonB5G Decision Engine"

[D18] MonB5G: "D6.1: Technical Report on System Integration and Operation"

[D19] MonB5G: " D6.2: Technical Report on the Integration of MonB5G Technologies in the Network Architecture"

[D20] ETHER: "D2.1: Initial report on ETHER network architecture, interfaces, and architecture evaluation"

[D21] ETHER: "D2.4: Final report on ETHER network architecture, interfaces, and architecture evaluation"

[D22] ETHER: "D4.1: Initial report on zero-touch data-driven network and service orchestration in the 3D ETHER architecture"

[D23] ETHER: "D4.2: Final report on zero-touch data-driven network and service orchestration in the 3D ETHER architecture"

[D24] 6G-Cloud: "D5.2: Overall framework PoC implementation"

7.6 Awards & distincions

- Participation in Industrial PhD programme (2020-2024)
- Best paper award for co-authored paper "Implementation-Aware AI Management Framework for AI-Native Networks", 2025 International Conference on 6G Networking (6GNet 2025)

Glossaries

3D Three-dimensional

3DQR 3D QoS Routing

3GPP 3rd Generation Partnership Project

4G 4th Generation

5G 5th Generation

5GS 5G System

5QI 5G QoS Identifier

6G 6th Generation

6G-UPF 6G User Plane Function

6GS 6G System

A2X Aircraft to Everything

AAM Advanced Air Mobility

AI Artificial Intelligence

ANUP Access Node User Plane

AP Application Plane

AR Augmented Reality

BN Border Nodes

C-SDN Centralized SDN

C2 Command and Control

CBRP Content-Based Routing Proxy

CN Core Network

CP Control Plane

CUPS Control and User Plane Separation

DDPG Deep Deterministic Policy Gradient

DDQN Double Deep Q-Network

DGA DRL-GNN Agent

DLBA Domain Load Balancing Agent

DN Data Network

DP Data Plane

DPI Deep Packet Inspection

DQN Deep Q-Network

DQR Domain QoS-aware Router

DRL Deep Reinforcement Learning

E2E End-to-End

eMBB Enhanced Mobile Broadband

EPC Evolved Packet Core

ETSI European Telecommunications Standards Institute

EU European Union

FL Feeder Link

GAP Global Attention Pooling

GBRSS Guaranteed Bit Rate Streaming Service

GLBA Global Load Balancing Agent

GNN Graph Neural Network

GSDNC Global SDNC

GTP GPRS Tunneling Protocol

GUTMA Global UTM Association

H-SDN Hierarchical multi-controller SDN

HDLLC High Data rate and Low Latency Communications

HDRL-LB Hierarchical Deep Reinforcement Learning Load Balancer

HMTC High-Performance Machine-Type Communications

HSDCN Hybrid SDN-based Data-Centric Core Network

HSDNC Hierarchical SDN Controller

ILP Integer Linear Programming

INF Intelligent NF

IoT Internet of Things

IP Internet Protocol

IPL Intelligence Plane

ISL Inter-Satellite Link

KPI Key Performance Indicator

KQI Key Quality Indicator

LEO Low Earth Orbit

LLM Large Language Model

M-QR Main QoS-aware Router

MADRL Multi-Agent Deep Reinforcement Learning

MANO Management and Orchestration

MAPE-K Monitor-Analyse-Plan-Execute based on Knowledge

MIoT Massive Internet of Things

ML Machine Learning

mMTC Massive Machine Type Communications

MNO Mobile Network Operator

MP Management Plane

MPLS-TE Multi-Protocol Label Switching Traffic Engineering

MPNN Message Passing Neural Network

MSHE Ministry of Science and Higher Education

mUPF master UPF

NBI NorthBound Interface

NDB Network Database

NDT Network Digital Twin

NF Network Function

NFV Network Function Virtualisation

NFV-MANO Network Function Virtualisation Management and Orchestration

NGMN Next Generation Mobile Networks

NOD Network Orchestration Domain

NS Network Slicing

NSI Network Slice Instance

NST Network Slice Template

NTN Non-Terrestrial Network

OF OpenFlow

OFDP OpenFlow Discovery Protocol

OSM Open-Source MANO

OSPF Open Shortest Path First

OSS/BSS Operations/Business Support System

OVS Open Virtual Switch

P4 Programming Protocol-Independent Packet Processors

PCF Policy and Charging Function

PFCP Packet Forwarding Control Protocol

PI Performance Indicators

PLMN Public Land Mobile Network

PM Performance Management

PoC Proof of Concept

QoE Quality of Experience

QoS Quality of Service

RAN Radio Access Network

RL Reinforcement Learning

RTT Round-Trip Time

SAC Slice Admission Control

SBA Service-Based Architecture

SCP Service Communication Proxy

SD-WAN Software-Defined Wide Area Network

SDN Software-Defined Networking

SDN-TE SDN Traffic Engineering

SDNC SDN Controller

SDO Standards Developing Organization

SDX Software-Defined Internet Exchange Point

SFC Service Function Chaining

SLA Service Level Agreement

SMF Session Management Function

SoA State of the Art

SOD Service Orchestration Domain

SP Shortest-Path

SRv6 Segment Routing over IPv6

SST Slice/Service Type

SUP-DTE SDN-UP with DRL-TE

sUPF sub-UPF

TASAC Time-of-Day-Aware Slice Admission Control

TCAM Ternary Content-Addressable Memory

TCP Transport Control Protocol

TE Traffic Engineering

TLE Two-Line Elements

TN Terrestrial Network

ToD Time-of-Day

TRL Technology Readiness Level

TTS Time-To-Serve

UAS NF Unmanned Aviation System Network Function

UAV Unmanned Aerial Vehicle

UDP User Datagram Protocol

UE User Equipment

UP User Plane

UPF User Plane Function

URLLC Ultra-Reliable Low-Latency Communication

UTM Unmanned Aircraft Systems Traffic Management

V2X Vehicle to Everything

VLAN Virtual Local Area Network

VNF Virtual Network Function

VR Virtual Reality

WD Weighted Dijkstra

Bibliography

[1] Ellie Zontou. *Unveiling the Evolution of Mobile Networks: From 1G to 7G*. Oct. 2023. DOI: 10.48550/arXiv.2310.19195. arXiv: 2310.19195 [cs]. (Visited on 05/06/2026).

[2] ITU-R. *Framework and Overall Objectives of the Future Development of IMT for 2030 and Beyond*. Recommendation M.2160-0 (11/2023). International Telecommunication Union – Radiocommunication Sector, Nov. 2023.

[3] Corici, Marius-Iulian and Franke, Norbert and Heyn, Thomas and Kontes, Georgios and Leyh, Martin and Magedanz, Thomas and Maaß, Uwe and Mikulla, Michael and Niemann, Bernhard and Peter, Michael and Roth-Mandutz, Elke and Schubert, Colja and Yammine, George. *On the Road to 6G: Drivers, Challenges and Enabling Technologies*. White Paper v1.0. Fraunhofer Fokus, 2021.

[4] Walid Saad, Mehdi Bennis, and Mingzhe Chen. "A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems". In: *IEEE Network* 34.3 (2020). doi: 10.1109/MNET.001.1900287, pp. 134–142.

[5] Emilio Calvanese Strinati et al. "6G: The Next Frontier: From Holographic Messaging to Artificial Intelligence Using Subterahertz and Visible Light Communication". In: *IEEE Vehicular Technology Magazine* 14.3 (Sept. 2019), pp. 42–50. ISSN: 1556-6080. DOI: 10.1109/MVT.2019.2921162. (Visited on 11/13/2024).

[6] 3GPP. *Study on Architecture for 6G System; Stage 2*. Technical Report TR 23.801, ver. 0.5.0. 3\textsuperscriptrd Generation Partnership Project, Apr. 2026.

[7] Othmane Blial, Mouad Ben Mamoun, and Redouane Benaini. "An Overview on SDN Architectures with Multiple Controllers". In: *Journal of Computer Networks and Communications* 2016.1 (2016), p. 9396525. ISSN: 2090-715X. DOI: 10.1155/2016/9396525. (Visited on 07/27/2025).

[8] Tamal Das, Vignesh Sridharan, and Mohan Gurusamy. "A Survey on Controller Placement in SDN". In: *IEEE Communications Surveys & Tutorials* 22.1 (2020), pp. 472–503. ISSN: 1553-877X, 2373-745X. DOI: 10.1109/COMST.2019.2935453. (Visited on 08/01/2025).

[9] Murat Karakus and Arjan Durresi. "A Survey: Control Plane Scalability Issues and Approaches in Software-Defined Networking (SDN)". In: *Computer Networks* 112 (Jan. 2017), pp. 279–293. ISSN: 13891286. DOI: 10.1016/j.comnet.2016.11.017. (Visited on 12/25/2021).

[10] Orange. *Mobile Network Technology Evolutions Beyond 2030*. White Paper. Orange, Apr. 2024. URL: https://hellofuture.orange.com/app/uploads/2024/05/2024-Orange-white-paper-on-Mobile-Network-Technology-Evolutions-Beyond-2030.pdf.

[11] 3GPP. *System Architecture for the 5G System (5GS)*. Technical Standard TS 23.501, ver. 19.2.1. 3rd Generation Partnership Project, Jan. 2025.

[12] 3GPP. *Study on 6G Use Cases and Service Requirements*. Technical Report TR 22.870, ver. 20.0.0. 3\textsuperscriptrd Generation Partnership Project, Mar. 2026.

[13] Zhaohui Zhang and Keyur Patel. *ANUP Implementation in 5G with BGP Signaling*. Internet-Draft draft-zzhang-dmm-anup5g-signaling-00. Work in Progress. July 2023. URL: https://www.ietf.org/archive/id/draft-zzhang-dmm-anup5g-signaling-00.html.

[14] Weiqiang Cheng et al. *Compressed SRv6 Segment List Encoding*. Internet-Draft draft-ietf-spring-srv6-srh-compression-19. Work in Progress. Nov. 2024. URL: https://datatracker.ietf.org/doc/html/draft-ietf-spring-srv6-srh-compression-19.

[15] Pier Luigi Ventre et al. "SDN Architecture and Southbound APIs for IPv6 Segment Routing Enabled Wide Area Networks". In: *arXiv preprint arXiv:1810.06008* (2018). arXiv: 1810.06008 [cs.NI]. URL: https://arxiv.org/abs/1810.06008.

[16] Kostas Pentikousis, Yan Wang, and Weihua Hu. "Dynamic Tunnel Switching for SDN-based Cellular Core Networks". In: *2014 IEEE International Conference on Communications (ICC)*. IEEE, 2014, pp. 2577–2582. DOI: 10.1109/ICC.2014.6883718. URL: https://www.researchgate.net/publication/266660632_Dynamic_tunnel_switching_for_SDN-based_cellular_core_networks.

[17] Arpit Gupta et al. "SDX: A Software Defined Internet Exchange". In: *Proceedings of the ACM SIGCOMM 2014 Conference*. SIGCOMM '14. ACM, 2014, pp. 551–562. DOI: 10.1145/2619239.2626300.

[18] ONF. *OpenFlow Switch Specification; Version 1.5.1 (Protocol Version 0x06)*. Specification ONF TS-025. Open Networking Foundation, Mar. 2015.

[19] Sushant Jain et al. "B4: Experience with a Globally-Deployed Software Defined WAN". In: *ACM SIGCOMM Computer Communication Review*. Vol. 43. Aug. 2013, pp. 3–14. DOI: 10.1145/2486001.2486019.

[20] Chi-Yao Hong et al. "Achieving High Utilization with Software-Driven WAN". In: *Proceedings of the ACM SIGCOMM 2013 Conference*. SIGCOMM '13. New York, NY, USA: ACM, 2013, pp. 15–26. DOI: 10.1145/2486001.2486012.

[21] Tao Huang et al. "A Survey on Large-Scale Software Defined Networking (SDN) Testbeds: Approaches and Challenges". In: *IEEE Communications Surveys & Tutorials* PP (Nov. 2016), pp. 1–1. DOI: 10.1109/COMST.2016.2630047.

[22] Abdelrahman Abuarqoub. "A Review of the Control Plane Scalability Approaches in Software Defined Networking". In: *Future Internet* 12.3 (2020). ISSN: 1999-5903. DOI: 10.3390/fi12030049.

[23] Murat Karakus and Arjan Durresi. "A survey: Control plane scalability issues and approaches in Software-Defined Networking (SDN)". In: *Computer Networks* 112 (2017). doi: 10.1016/j.comnet.2016.11.017, pp. 279–293. ISSN: 1389-1286. DOI: 10.1016/j.comnet.2016.11.017.

[24] Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Evaluating Scalability of Hierarchical Multi-controller SDN: A Holistic Approach". In: *Journal of Network and Systems Management* 34.4 (May 2026), p. 107. ISSN: 1573-7705. DOI: 10.1007/s10922-026-10087-w.

[25] Robert Kołakowski, Slawomir Kukliński, and Lechosław Tomaszewski. "Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Networks". In: *2024 IFIP Networking Conference (IFIP Networking)*. June 2024, pp. 607–612. DOI: 10.23919/IFIPNetworking62109.2024.10619899.

[26] Robert Kołakowski et al. "Hierarchical Traffic Engineering in 3D Networks Using QoS-Aware Graph-Based Deep Reinforcement Learning". In: *Electronics* 14.5 (Mar. 2025), p. 1045. ISSN: 2079-9292. DOI: 10.3390/electronics14051045.

[27] Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Time-of-Day-Aware Slice Admission Control". In: *2023 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. Sept. 2023, pp. 199–204. DOI: 10.1109/MeditCom58224.2023.10266645.

[28] Lechosław Tomaszewski and Robert Kołakowski. "Advanced Air Mobility and Evolution of Mobile Networks". In: *Drones* 7.9 (Sept. 2023), p. 556. ISSN: 2504-446X. DOI: 10.3390/drones7090556. (Visited on 02/27/2025).

[29] Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "On 6G-Enabled SDN-Based Mobile Network User Plane with DRL-Based Traffic Engineering". In: *Artificial Intelligence Applications and Innovations. AIAI 2024 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2024, pp. 124–138. ISBN: 978-3-031-63227-3. DOI: 10.1007/978-3-031-63227-3_9.

[30] Robert Kołakowski, Thierry Lejkin, and Victorien Romain. "Hybrid SDN-Based Data-Centric Mobile Core Network with Content-Based Routing". In: *2025 International Conference on 6G Networking (6GNet 2025)*. Paris, France, Dec. 2025, pp. 126–130. DOI: 10.1109/6GNet68413.2025.11314080.

[31] Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "Performance evaluation of the OSM orchestrator". In: *2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. Nov. 2021, pp. 15–20. DOI: 10.1109/NFV-SDN53031.2021.9665052.

[32] José Gómez-delaHiz and Jaime Galán-Jiménez. "Improving the Traffic Engineering of SDN Networks by Using Local Multi-Agent Deep Reinforcement Learning". In: *NOMS 2024-2024 IEEE Network Operations and Management Symposium*. Seoul, Korea, Republic of: IEEE, May 2024, pp. 1–5. ISBN: 9798350327939. DOI: 10.1109/NOMS59830.2024.10575465. (Visited on 08/03/2025).

[33] Tariq Emad Ali, Csaba Biro, and Dhulfiqar Zoltan Alwahab. "A Proposed Method for TCAM Optimization in SDN". In: *Journal of Computer Networks and Communications* 2026.1 (2025). ISSN: 2090-715X. DOI: 10.1155/jcnc/4202579.

[34] Petar D. Bojović et al. "Dynamic QoS Management for a Flexible 5G/6G Network Core: A Step toward a Higher Programmability". In: *Sensors* 22.8 (Jan. 2022), p. 2849. ISSN: 1424-8220. DOI: 10.3390/s22082849. (Visited on 04/15/2025).

[35] Van Sy Mai et al. *End-to-End Quality-of-Service Assurance with Autonomous Systems: 5G/6G Case Study*. Jan. 2022. DOI: 10.48550/arXiv.2201.13300. arXiv: 2201.13300 [eess]. (Visited on 04/15/2025).

[36] Arusa Kanwal et al. "Exploring Security Dynamics in SDN Controller Architectures: Threat Landscape and Implications". In: *IEEE Access* 12 (2024), pp. 56517–56553. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2024.3390968. (Visited on 08/01/2025).

[37] Haythem Yahyaoui and Mohamed Faten Zhani. "On Providing Low-cost Flow Monitoring for SDN Networks". In: *2020 IEEE 9th International Conference on Cloud Networking (CloudNet)*. Piscataway, NJ, USA: IEEE, Nov. 2020, pp. 1–6. ISBN: 978-1-72819-486-8. DOI: 10.1109/CloudNet51028.2020.9335797. (Visited on 04/09/2025).

[38] Kokouvi Benoit Nougnanke, Marc Bruyere, and Yann Labit. "Low-Overhead Near-Real-Time Flow Statistics Collection in SDN". In: *2020 6th IEEE Conference on Network Softwarization (NetSoft)*. Ghent, Belgium: IEEE, June 2020, pp. 155–159. ISBN: 978-1-72815-684-2. DOI: 10.1109/NetSoft48620.2020.9165421. (Visited on 04/09/2025).

[39] Ali Malik and Ruairí De Fréin. "Graph Modeling for OpenFlow Switch Monitoring". In: *IEEE Access* 11 (2023), pp. 84543–84553. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3303847. (Visited on 05/09/2025).

[40] Abdelhadi Azzouni et al. "Limitations of Openflow Topology Discovery Protocol". In: *2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*. Budva, Montenegro: IEEE, June 2017, pp. 1–3. ISBN: 978-1-5386-2077-9. DOI: 10.1109/MedHocNet.2017.8001642. (Visited on 04/09/2025).

[41] Raniyah Wazirali, Rami Ahmad, and Suheib Alhiyari. "SDN-OpenFlow Topology Discovery: An Overview of Performance Issues". In: *Applied Sciences* 11.15 (Jan. 2021), p. 6999. ISSN: 2076-3417. DOI: 10.3390/app11156999. (Visited on 04/09/2025).

[42] S. H. Yeganeh, A. Tootoonchian, and Y. Ganjali. "On Scalability of Software-Defined Networking". In: *IEEE Communications Magazine* 51.2 (Feb. 2013), pp. 136–141. ISSN: 0163-6804. DOI: 10.1109/MCOM.2013.6461198. (Visited on 02/03/2025).

[43] Qingyun Zuo et al. "On Generality of the Data Plane and Scalability of the Control Plane in Software-Defined Networking". In: *China Communications* 11.2 (Feb. 2014), pp. 55–64. ISSN: 1673-5447. DOI: 10.1109/CC.2014.6821737. (Visited on 02/03/2025).

[44] Murat Karakus and Arjan Durresi. "A Scalability Metric for Control Planes in Software Defined Networks (SDNs)". In: *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*. Crans-Montana, Switzerland: IEEE, Mar. 2016, pp. 282–289. ISBN: 978-1-5090-1858-1. DOI: 10.1109/AINA.2016.112. (Visited on 02/03/2025).

[45] Firas Zobary and Chun Lin Li. "A Mathematical Model for SDN Control Plane Scalability Evaluation Based on Controller Utilization". In: *Journal of Control Engineering and Applied Informatics* 25 (2023). DOI: 10.61416/ceai.v25i4.8783. (Visited on 01/31/2025).

[46] Mohamed A. Aglan, Mohamed A. Sobh, and Ayman M. Bahaa-Eldin. "Reliability and Scalability in SDN Networks". In: *2018 13th International Conference on Computer Engineering and Systems (ICCES)*. Cairo, Egypt: IEEE, Dec. 2018, pp. 549–554. ISBN: 978-1-5386-5111-7. DOI: 10.1109/ICCES.2018.8639201. (Visited on 02/03/2025).

[47] Jie Hu et al. "Scalability of Control Planes for Software Defined Networks: Modeling and Evaluation". In: *2014 IEEE 22nd International Symposium of Quality of Service (IWQoS)*. Hong Kong: IEEE, May 2014, pp. 147–152. ISBN: 978-1-4799-4852-9. DOI: 10.1109/IWQoS.2014.6914314. (Visited on 11/02/2023).

[48] Jianglong Wang et al. "A Multi-Domain SDN Scalability Architecture Implementation Based on the Coordinate Controller". In: *2016 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*. Chengdu, China: IEEE, Oct. 2016, pp. 494–499. ISBN: 978-1-5090-5154-0. DOI: 10.1109/CyberC.2016.100. (Visited on 02/03/2025).

[49] Luca Ballotta, Mihailo R. Jovanović, and Luca Schenato. "Can Decentralized Control Outperform Centralized? The Role of Communication Latency". In: *IEEE Transactions on Control of Network Systems* 10.3 (Sept. 2023), pp. 1629–1640. ISSN: 2325-5870, 2372-2533. DOI: 10.1109/TCNS.2023.3237483. arXiv: 2109.00359 [math]. (Visited on 05/09/2025).

[50] Hemanth Kumar Ravuri et al. "On Routing Scalability in Flat SDN Architectures". In: *2020 11th International Conference on Network of the Future (NoF)*. Bordeaux, France: IEEE, Oct. 2020, pp. 23–27. ISBN: 978-1-72818-055-7. DOI: 10.1109/NoF50125.2020.9249170. (Visited on 02/03/2025).

[51] Teemu Koponen et al. “Onix: A Distributed Control Platform for Large-Scale Production Networks”. In: *Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation*. OSDI’10. USA: USENIX Association, Oct. 2010, pp. 351–364. (Visited on 07/27/2025).

[52] Kévin Phemius, Mathieu Bouet, and Jérémie Leguay. “DISCO: Distributed SDN Controllers in a Multi-Domain Environment”. In: *2014 IEEE Network Operations and Management Symposium (NOMS)*. May 2014, pp. 1–2. DOI: 10.1109/NOMS.2014.6838273. (Visited on 08/24/2025).

[53] Advait Dixit et al. “Towards an Elastic Distributed SDN Controller”. In: *SIGCOMM Comput. Commun. Rev.* 43.4 (Aug. 2013), pp. 7–12. ISSN: 0146-4833. DOI: 10.1145/2534169.2491193. (Visited on 07/27/2025).

[54] Yonghong Fu et al. “Orion: A Hybrid Hierarchical Control Plane of Software-Defined Networking for Large-Scale Networks”. In: *2014 IEEE 22nd International Conference on Network Protocols*. Oct. 2014, pp. 569–576. DOI: 10.1109/ICNP.2014.91. (Visited on 07/27/2025).

[55] Soheil Hassas Yeganeh and Yashar Ganjali. “Kandoo: A Framework for Efficient and Scalable Offloading of Control Applications”. In: *Proceedings of the First Workshop on Hot Topics in Software Defined Networks*. Helsinki Finland: ACM, Aug. 2012, pp. 19–24. ISBN: 978-1-4503-1477-0. DOI: 10.1145/2342441.2342446. (Visited on 07/27/2025).

[56] Zheng Cai, Alan L Cox, and T S Eugene Ng. “Maestro: A System for Scalable OpenFlow Control”. In: *Computer Science Technical Reports* (Apr. 2010).

[57] Aditi SR et al. “OpenDaylight as Software Defined Networking Controller: Shortcomings and Possible Solutions”. In: *2022 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. July 2022, pp. 1–6. DOI: 10.1109/CONECCT55679.2022.9865784. (Visited on 07/27/2025).

[58] Pankaj Berde et al. “ONOS: Towards an Open, Distributed SDN OS”. In: *Proceedings of the Third Workshop on Hot Topics in Software Defined Networking*. HotSDN ’14. New York, NY, USA: Association for Computing Machinery, Aug. 2014, pp. 1–6. ISBN: 978-1-4503-2989-7. DOI: 10.1145/2620728.2620744. (Visited on 07/27/2025).

[59] Rui Kubo et al. “Ryu SDN Framework-Open-Source SDN Platform Software”. In: *NTT Technical Review* 12 (Aug. 2014). DOI: 10.53829/ntr201408fa4. (Visited on 08/04/2025).

[60] Shimin Sun et al. “Load Balancing Algorithm for SDN Multi-Controller with Elite Genetic Algorithm”. In: *2023 IEEE 6th International Conference on Electronics and Communication Engineering (ICECE)*. Xi’an, China: IEEE, Dec. 2023, pp. 146–150. ISBN: 9798350328721. DOI: 10.1109/ICECE59822.2023.10462299. (Visited on 11/11/2025).

[61] Hamza Mokhtar et al. "Dynamic Controller Assignment in LEO Satellite Based SDN Using Multi-Agent Deep Reinforcement Learning". In: *2024 IEEE Conference on Standards for Communications and Networking (CSCN)*. Belgrade, Serbia: IEEE, Nov. 2024, pp. 208–211. ISBN: 9798331507428. DOI: 10.1109/CSCN63874.2024.10849764. (Visited on 11/11/2025).

[62] Haythem Yahyaoui et al. "On minimizing flow monitoring costs in large-scale software-defined network networks". In: *International Journal of Network Management* 33.2 (2023), e2220. DOI: https://doi.org/10.1002/nem.2220. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nem.2220. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/nem.2220.

[63] Min-Han Hung et al. "A SDN Controller Monitoring Architecture for 5G Backhaul Networks". In: *2022 23rd Asia-Pacific Network Operations and Management Symposium (APNOMS)*. Takamatsu, Japan: IEEE, Sept. 2022, pp. 1–4. ISBN: 978-4-88552-339-7. DOI: 10.23919/APNOMS56106.2022.9919988. (Visited on 03/10/2026).

[64] Kanwal Saeed and Muhammad Obaid Ullah. "Toward Reliable Controller Placements in Software-Defined Network Using Constrained Multi-Objective Optimization Technique". In: *IEEE Access* 10 (2022), pp. 129865–129883. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2022.3228039. (Visited on 11/11/2025).

[65] Tomofumi Kondo et al. "Energy-Aware Controller Placement With Delay Guarantees for In-Band SDN". In: *International Journal of Network Management* 36.1 (2026), e70036. ISSN: 1099-1190. DOI: 10.1002/nem.70036. (Visited on 03/10/2026).

[66] Tamal Das and Mohan Gurusamy. "Controller Placement for Resilient Network State Synchronization in Multi-Controller SDN". In: *IEEE Communications Letters* 24.6 (June 2020), pp. 1299–1303. ISSN: 1089-7798, 1558-2558, 2373-7891. DOI: 10.1109/LCOMM.2020.2979072. (Visited on 11/11/2025).

[67] Ilora Maity, Konstantinos Ntontin, and Symeon Chatzinotas. "CoMOSat: Controller Placement in SDN-Enabled Multi-Orbit Satellite Networks". In: *2025 IEEE International Conference on Communications Workshops (ICC Workshops)*. Montreal, QC, Canada: IEEE, June 2025, pp. 2026–2032. ISBN: 9798331596248. DOI: 10.1109/ICCWorkshops67674.2025.11162452. (Visited on 11/11/2025).

[68] F. Paolucci et al. "P4 Edge Node Enabling Stateful Traffic Engineering and Cyber Security". In: *Journal of Optical Communications and Networking* 11.1 (Jan. 2019), A84–A95. ISSN: 1943-0639. (Visited on 08/03/2025).

[69] Elie F. Kfoury, Jorge Crichigno, and Elias Bou-Harb. "An Exhaustive Survey on P4 Programmable Data Plane Switches: Taxonomy, Applications, Challenges, and Future Trends". In: *IEEE Access* 9 (2021), pp. 87094–87155. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2021.3086704. (Visited on 08/18/2025).

[70] Chanbin Bae et al. “Quantized In-band Network Telemetry for Low Bandwidth Overhead Monitoring”. In: *2024 20th International Conference on Network and Service Management (CNSM)*. Prague, Czech Republic: IEEE, Oct. 2024, pp. 1–5. ISBN: 978-3-903176-66-9. DOI: 10.23919/CNSM62983.2024.10814621. (Visited on 08/03/2025).

[71] Nitin Varyani, Zhi-Li Zhang, and David Dai. “QROUTE: An Efficient Quality of Service (QoS) Routing Scheme for Software-Defined Overlay Networks”. In: *IEEE Access* 8 (2020), pp. 104109–104126. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2995558. (Visited on 08/03/2025).

[72] ETSI. *TeraFlowSDN*. https://www.teraflow-h2020.eu/. (Visited on 07/27/2025).

[73] Abdelrahman Abuarqoub. “A Review of the Control Plane Scalability Approaches in Software Defined Networking”. In: *Future Internet* 12.3 (Mar. 2020), p. 49. ISSN: 1999-5903. DOI: 10.3390/fi12030049. (Visited on 08/24/2025).

[74] Dana Turlykozhayeva et al. “Routing Algorithm for Software Defined Network Based on Boxcovering Algorithm”. In: *2023 10th International Conference on Wireless Networks and Mobile Communications (WINCOM)*. Oct. 2023, pp. 1–5. DOI: 10.1109/WINCOM59760.2023.10322960. arXiv: 2406.05528 [cs]. (Visited on 08/24/2025).

[75] Zequn Jia et al. “cRetor: An SDN-Based Routing Scheme for Data Centers With Regular Topologies”. In: *IEEE Access* 8 (2020), pp. 116866–116880. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.3004609. (Visited on 08/24/2025).

[76] Paul E. Wright. “Two Parallel Processors with Coupled Inputs”. In: *Advances in Applied Probability* 24.4 (Dec. 1992), pp. 986–1007. ISSN: 0001-8678, 1475-6064. DOI: 10.2307/1427722. (Visited on 07/20/2025).

[77] Bob Lantz, Brandon Heller, and Nick McKeown. “A Network in a Laptop: Rapid Prototyping for Software-Defined Networks”. In: *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*. Hotnets-IX. New York, NY, USA: Association for Computing Machinery, Oct. 2010, pp. 1–6. ISBN: 978-1-4503-0409-2. DOI: 10.1145/1868447.1868466. (Visited on 08/24/2025).

[78] Linux Foundation. *Open vSwitch*. https://www.openvswitch.org/. (Visited on 08/21/2024).

[79] K. Ntontin et al. “ETHER: A 6G Architectural Framework for 3D Multi-Layered Networks”. In: *2024 IEEE Wireless Communications and Networking Conference (WCNC)*. Dubai, United Arab Emirates: IEEE, Apr. 2024, pp. 1–6. ISBN: 9798350303582. DOI: 10.1109/WCNC57260.2024.10570731. (Visited on 02/07/2025).

[80] Chen Chen et al. “Hierarchical Domain-Based Multicontroller Deployment Strategy in SDN-Enabled Space–Air–Ground Integrated Network”. In: *IEEE Transactions on Aerospace and Electronic Systems* 58.6 (Dec. 2022), pp. 4864–4879. ISSN: 0018-9251, 1557-9603, 2371-9877. DOI: 10.1109/TAES.2022.3199191. (Visited on 02/07/2025).

[81] Zhan Liao et al. “Multi-Controller Deployment in SDN-Enabled 6G Space–Air–Ground Integrated Network”. In: *Remote Sensing* 14.5 (Jan. 2022), p. 1076. ISSN: 2072-4292. DOI: 10.3390/rs14051076. (Visited on 02/07/2025).

[82] Slawomir Kuklinski, Yuhong Li, and Khoa Truong Dinh. “Handover management in SDN-based mobile networks”. In: *2014 IEEE Globecom Workshops (GC Wkshps)*. Austin, TX, USA: IEEE, Dec. 2014, pp. 194–200. ISBN: 978-1-4799-7470-2. DOI: 10.1109/GLOCOMW.2014.7063430. (Visited on 03/12/2026).

[83] Anwar Elwalid et al. *Overview and Principles of Internet Traffic Engineering*. Request for Comments RFC 3272. Internet Engineering Task Force, May 2002. (Visited on 08/05/2024).

[84] Andrea Baiocchi. *Network Traffic Engineering: Stochastic Models and Applications*. 1st ed. Wiley, Aug. 2020. ISBN: 978-1-119-63243-6. (Visited on 08/05/2024).

[85] Ian F. Akyildiz et al. “A roadmap for traffic engineering in SDN-OpenFlow networks”. In: *Computer Networks* 71 (Oct. 2014), pp. 1–30. ISSN: 13891286. DOI: 10.1016/j.comnet.2014.06.002. (Visited on 08/06/2024).

[86] Alaitz Mendiola et al. “A Survey on the Contributions of Software-Defined Networking to Traffic Engineering”. In: *IEEE Communications Surveys & Tutorials* 19.2 (2017), pp. 918–953. ISSN: 1553-877X. DOI: 10.1109/COMST.2016.2633579. (Visited on 12/25/2021).

[87] Mohammed Salman, Abdulrahman Alenezi, and Ahmed Alzahrani. “Security-Enhanced Traffic Engineering Frameworks Against DDoS Attacks on Massive Data Flows in Software-Defined Infrastructures”. In: *IEEE Access* 12 (2024). DOI: 10.1109/ACCESS.2024.3438360. URL: https://www.researchgate.net/publication/404915563_Security-Enhanced_Traffic_Engineering_Frameworks_Against_DDoS_Attacks_on_Massive_Data_Flows_in_Software-Defined_Infrastructures.

[88] 3GPP. *Study on Using Satellite Access in 5G*. Technical Report TR 22.822, ver. 16.0.0. 3rd Generation Partnership Project, Dec. 2018. (Visited on 11/13/2024).

[89] Alessandro Guidotti et al. “The Path to 5G-Advanced and 6G Non-Terrestrial Network Systems”. In: *2022 11th Advanced Satellite Multimedia Systems Conference and the 17th Signal Processing for Space Communications Workshop (ASMS/SPSC)*. Graz, Austria: IEEE, Sept. 2022, pp. 1–8. ISBN: 978-1-66547-377-4. DOI: 10.1109/ASMS/SPSC55670.2022.9914764. (Visited on 11/13/2024).

[90] Lechosław Tomaszewski et al. “ETHER: Energy- and Cost-Efficient Framework for Seamless Connectivity over the Integrated Terrestrial and Non-terrestrial 6G Networks”. In: *Artificial Intelligence Applications and Innovations. AIAI 2023 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. León, Spain: Springer Nature Switzerland, June 2023, pp. 32–44. ISBN: 978-3-031-34171-7. DOI: 10.1007/978-3-031-34171-7_2.

[91] ITU-R. *Future Technology Trends of Terrestrial International Mobile Telecommunications Systems towards 2030 and Beyond*. Report M.2516-0. International Telecommunication Union – Radiocommunication Sector, Nov. 2022.

[92] Daniel King and Kevin Shortt. *Time Variant Challenges for Non-Terrestrial Networks*. Internet Draft draft-king-tvr-ntn-challanges-00. Internet Engineering Task Force, Jan. 2023. (Visited on 11/03/2024).

[93] I. Ali, N. Al-Dhahir, and J.E. Hershey. “Predicting the Visibility of LEO Satellites”. In: *IEEE Transactions on Aerospace and Electronic Systems* 35.4 (Oct. 1999), pp. 1183–1190. ISSN: 00189251. DOI: 10.1109/7.805436. (Visited on 11/13/2024).

[94] IETF NWG. *OSPF Version 2*. Tech. rep. IETF, 1998. URL: https://www.ietf.org/rfc/rfc2328.txt.

[95] Giovanni Geraci et al. “Integrating Terrestrial and Non-terrestrial Networks: 3D Opportunities and Challenges”. In: 61.4 (July 2022), pp. 42–48. DOI: 10.1109/MCOM.002.2200366.

[96] Lin Han et al. “Large Scale LEO Satellite Networks for the Future Internet: Challenges and Solutions to Addressing and Routing”. In: *Computer Networks and Communications* (Dec. 2022). ISSN: 2972-4619. DOI: 10.37256/cnc.1120232105. (Visited on 11/14/2024).

[97] Pengyuan Du et al. “Multipath TCP in SDN-enabled LEO Satellite Networks”. In: *MILCOM 2016 – 2016 IEEE Military Communications Conference*. Baltimore, MD, USA: IEEE, Nov. 2016, pp. 354–359. ISBN: 978-1-5090-3781-0. DOI: 10.1109/MILCOM.2016.7795352. (Visited on 11/13/2024).

[98] Victor Monzon Baeza et al. “IoT-NTN Communications via Store-and-Forward Core Network in Multi-LEO-satellite Deployments”. In: *2024 IEEE 35th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. Valencia, Spain: IEEE ComSoc, Sept. 2024, pp. 1–6. DOI: 10.1109/PIMRC59610.2024.10817343.

[99] 3GPP. *Study on Satellite Access – Phase 3*. Technical Report TR 22.865, ver. 19.2.0. 3rd Generation Partnership Project, Dec. 2023. (Visited on 11/13/2024).

[100] Abdelrahman Abuarqoub. “A Review of the Control Plane Scalability Approaches in Software Defined Networking”. In: *Future Internet* 12.3 (2020). doi: 10.3390/fi12030049. ISSN: 1999-5903. DOI: 10.3390/fi12030049.

[101] Yuxiang Hu et al. “EARS: Intelligence-driven experiential network architecture for automatic routing in software-defined networking”. In: *China Communications* 17.2 (2020). doi: 10.23919/JCC.2020.02.013, pp. 149–162. DOI: 10.23919/JCC.2020.02.013.

[102] Daniela M. Casas-Velasco, Oscar Mauricio Caicedo Rendon, and Nelson L. S. da Fonseca. "Intelligent Routing Based on Reinforcement Learning for Software-Defined Networking". In: *IEEE Transactions on Network and Service Management* 18.1 (2021), pp. 870–881. DOI: 10.1109/TNSM.2020.3036911.

[103] Junyan Chen et al. "ALBRL: Automatic Load-Balancing Architecture Based on Reinforcement Learning in Software-Defined Networking". In: *Wireless Communications and Mobile Computing* 2022 (May 2022). doi: 10.1155/2022/3866143, pp. 1–17. DOI: 10.1155/2022/3866143.

[104] Juan Chafla Altamirano et al. "QoS-aware Network Self-management Architecture Based on DRL and SDN for Remote Areas". In: *2022 IEEE 11th IFIP International Conference on Performance Evaluation and Modeling in Wireless and Wired Networks (PEMWN)*. Rome, Italy: IEEE, Nov. 2022, pp. 1–6. ISBN: 978-3-903176-50-8. DOI: 10.23919/PEMWN56085.2022.9963841. (Visited on 08/06/2024).

[105] Xiao Chen et al. "A Distributed Routing Algorithm for LEO Satellite Networks: A Multi-Agent Transformer-MIX Learning Approach". In: *IEEE Internet of Things Journal* (2025), pp. 1–1. ISSN: 2327-4662, 2372-2541. DOI: 10.1109/JIOT.2025.3530919. (Visited on 02/19/2025).

[106] Hailong Liao et al. "Real-Time Routing Design for LEO Satellite Networks: An Enhanced Multi-Agent DRL Approach". In: *2024 IEEE/CIC International Conference on Communications in China (ICCC Workshops)*. Hangzhou, China: IEEE, Aug. 2024, pp. 547–552. ISBN: 9798350377675. DOI: 10.1109/ICCCWorkshops62562.2024.10693714. (Visited on 02/19/2025).

[107] Xiaoying Liu et al. "Distributed Computation Offloading for Energy Provision Minimization in WP-MEC Networks with Multiple HAPs". In: *IEEE Transactions on Mobile Computing* (2024), pp. 1–17. ISSN: 1558-0660. DOI: 10.1109/TMC.2024.3502004. (Visited on 02/26/2025).

[108] Yongyi Ran et al. "Fully-Distributed Dynamic Packet Routing for LEO Satellite Networks: A GNN-Enhanced Multi-Agent Reinforcement Learning Approach". In: *IEEE Transactions on Vehicular Technology* (2024), pp. 1–6. ISSN: 0018-9545, 1939-9359. DOI: 10.1109/TVT.2024.3499933. (Visited on 02/19/2025).

[109] Jian-Jhong Huang et al. "Weighted routing in hierarchical multi-domain SDN controllers". In: *2015 17th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. doi: 10.1109/APNOMS.2015.7275362. 2015, pp. 356–359. DOI: 10.1109/APNOMS.2015.7275362.

[110] Tarik Moufakir et al. "Collaborative Multi-domain Routing in SDN Environments". In: *Journal of Network and Systems Management* 30 (Jan. 2022). doi: 10.1007/s10922-021-09638-0. DOI: 10.1007/s10922-021-09638-0.

[111] Yangyang Liu et al. “Distributed Traffic Engineering for Multi-Domain SDN Without Trust”. In: *IEEE Transactions on Cloud Computing* 10.4 (2022). doi: 10.1109/TCC.2021.3067456, pp. 2481–2496. DOI: `10.1109/TCC.2021.3067456`.

[112] Miao Ye et al. “A New Intelligent Cross-Domain Routing Method in SDN Based on a Proposed Multiagent Reinforcement Learning Algorithm”. In: (2023). doi: 10.48550/arXiv.2303.07572. DOI: `10.48550/arXiv.2303.07572`. arXiv: `2303.07572 [cs.NI]`.

[113] Timothy P. Lillicrap et al. *Continuous control with deep reinforcement learning*. doi: 10.48550/arXiv.1509.0 2019. DOI: `10.48550/arXiv.1509.02971`. arXiv: `1509.02971 [cs.LG]`.

[114] Hado van Hasselt, Arthur Guez, and David Silver. “Deep Reinforcement Learning with Double Q-learning”. In: *30^{th} AAAI Conference on Artificial Intelligence*. Phoenix, AZ, USA, Sept. 2015. DOI: `10.1609/aaai.v30i1.10295`.

[115] *GN4-3N - GÉANT Network*. https://network.geant.org/gn4-3n/. Dec. 2020. (Visited on 07/28/2024).

[116] *Keras*. Last accessed 10 May 2024. URL: `https://keras.io/`.

[117] 3GPP. *System Architecture for the 5G System (5GS)*. Technical Standard TS 23.501, ver. 19.2.1. 3^{rd} Generation Partnership Project, Jan. 2025.

[118] Justin Gilmer et al. “Neural Message Passing for Quantum Chemistry”. In: *Proceedings of the 34^{th} International Conference on Machine Learning – Volume 70*. ICML’17. Sydney, NSW, Australia: JMLR.org, June 2017, pp. 1263–1272. DOI: `10.5555/3305381.3305512`.

[119] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. *Self-Attention Graph Pooling*. June 2019. DOI: `10.48550/arXiv.1904.08082`. (Visited on 02/26/2025).

[120] *SimPy Documentation*. `https://simpy.readthedocs.io/en/latest/`. (Visited on 11/13/2024).

[121] *PyG Documentation – Pytorch_geometric Documentation*. `https://pytorch-geometric.readthedocs.io/en/latest/`. (Visited on 11/13/2024).

[122] *Skyfield – Documentation*. `https://rhodesmill.org/skyfield/`. (Visited on 11/13/2024).

[123] *CelesTrak: Current Supplemental GP Element Sets*. `https://celestrak.org/NORAD/elements/supplemental/`. (Visited on 11/13/2024).

[124] Neil J. Gunther. “A Simple Capacity Model of Massively Parallel Transaction Systems”. In: *19^{th} International Computer Measurement Group Conference*. San Diego, CA, USA, 1993. (Visited on 12/10/2024).

[125] Mourice O. Ojijo and Olabisi E. Falowo. “A Survey on Slice Admission Control Strategies and Optimization Schemes in 5G Network”. In: *IEEE Access* 8 (Jan. 2020). doi: 10.1109/ACCESS.2020.29676 pp. 14977–14990. ISSN: 2169-3536. DOI: `10.1109/ACCESS.2020.2967626`.

[126] Hoang Duy Trinh et al. "Analysis and Modeling of Mobile Traffic Using Real Traces". In: *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. doi: 10.1109/PIMRC.2017.8292200. Montreal, QC: IEEE, Oct. 2017, pp. 1–6. ISBN: 978-1-5386-3529-2. DOI: 10.1109/PIMRC.2017.8292200.

[127] Shuo Wang et al. "An Approach for Spatial-temporal Traffic Modeling in Mobile Cellular Networks". In: *2015 27th International Teletraffic Congress* (Sept. 2015). doi: 10.1109/ITC.2015.31, pp. 203–209. DOI: 10.1109/ITC.2015.31. arXiv: 1703.10804.

[128] Vijayalakshmi Chetlapalli, K. S. S. Iyer, and Himanshu Agrawal. "Modelling Time-Dependent Aggregate Traffic in 5G Networks". In: *Telecommunication Systems* 73.4 (Apr. 2020). doi: 10.1007/s11235-019-00629-w, pp. 557–575. ISSN: 1572-9451. DOI: 10.1007/s11235-019-00629-w.

[129] William Fernando Villota Jácome, Oscar Mauricio Caicedo Rendon, and Nelson Luis Saldanha da Fonseca. *Admission Control for 5G Network Slicing Based on (Deep) Reinforcement Learning*. Preprint. doi: 10.36227/techrxiv.14498190. Apr. 2021. DOI: 10.36227/techrxiv.14498190.

[130] Sihem Bakri, Bouziane Brik, and Adlen Ksentini. "On Using Reinforcement Learning for Network Slice Admission Control in 5G: Offline vs. Online". In: *International Journal of Communication Systems* 34.7 (May 2021). doi: 10.1002/dac.4757, pp. 1–12. ISSN: 1074-5351, 1099-1131. DOI: 10.1002/dac.4757.

[131] Vincenzo Sciancalepore, Xavier Costa-Perez, and Albert Banchs. "RL-NSB: Reinforcement Learning-Based 5G Network Slice Broker". In: *IEEE/ACM Transactions on Networking* 27.4 (Aug. 2019). doi: 10.1109/TNET.2019.2924471, pp. 1543–1557. ISSN: 1063-6692, 1558-2566. DOI: 10.1109/TNET.2019.2924471.

[132] Bin Han, Di Feng, and Hans D. Schotten. "A Markov Model of Slice Admission Control". In: *IEEE Networking Letters* 1.1 (Mar. 2019). doi: 10.1109/LNET.2018.2873978, pp. 2–5. ISSN: 2576-3156. DOI: 10.1109/LNET.2018.2873978. arXiv: 1804.01861.

[133] Dario Bega et al. "A Machine Learning Approach to 5G Infrastructure Market Optimization". In: *IEEE Transactions on Mobile Computing* 19.3 (Mar. 2020). doi: 10.1109/TMC.2019.2896950, pp. 498–512. DOI: 10.1109/TMC.2019.2896950.

[134] Muhammad Rehan Raza et al. "Reinforcement Learning for Slicing in a 5G Flexible RAN". In: *Journal of Lightwave Technology* 37.20 (Oct. 2019). doi: 10.1109/JLT.2019.2924345, pp. 5161–5169. DOI: 10.1109/JLT.2019.2924345.

[135] Christopher J. C. H. Watkins and Peter Dayan. "Q-learning". In: *Machine Learning* 8 (May 1992). doi: 10.1007/BF00992698, pp. 279–292. DOI: 10.1007/BF00992698.

[136] Volodymyr Mnih et al. *Playing Atari with Deep Reinforcement Learning*. doi: 10.48550/arXiv.1312.5602. 2013. DOI: 10.48550/arXiv.1312.5602.

[137] PwC Drone Powered Solutions. *Advanced Air Mobility: From Concept to Commercial Reality*. Executive Outlook Report. An Executive Outlook for Airlines and Airports. PricewaterhouseCoopers (PwC), 2025. URL: https://cee.pwc.com/pdf-nf/PwC_DPS_Global_AAM_Report.pdf.

[138] NASA. *Advanced Air Mobility Project*. [Website]. https://www.nasa.gov/aeroresearch/programs/iasp/aam/description/ Accessed: 07 August, 2023.

[139] GUTMA. *UAS Traffic Management Architecture*. Ver. 1.0. https://www.gutma.org/docs/Global_UTM_Architecture_V1.pdf. Global UTM Association: Lausanne, Switzerland, Apr. 2017.

[140] 3GPP. *Study on Remote Identification of Unmanned Aerial Systems (UAS)*. en. Technical Report TR 22.825, ver. 16.0.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3527. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Sept. 2018.

[141] 3GPP. *Unmanned Aerial System (UAS) support in 3GPP*. en. Technical Standard TS 22.125, ver. 17.6.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3545. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Apr. 2022.

[142] 3GPP. *Enhancement for Unmanned Aerial Vehicles (UAVs)*. en. Technical Report TR 22.829, ver. 17.1.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3557. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Sept. 2019.

[143] 3GPP. *Study on Uncrewed Aerial Vehicle (UAV) Phase 3 (Release 19)*. en. Technical Report TR 22.843, ver. 19.0.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4090. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, June 2023.

[144] 3GPP. *Study on supporting Unmanned Aerial Systems (UAS) connectivity, identification and tracking*. en. Technical Report TR 23.754, ver. 17.1.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3575. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Mar. 2021.

[145] 3GPP. *Study on application layer support for Unmanned Aerial Systems (UAS)*. en. Technical Report TR 23.755, ver. 17.0.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3588. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Apr. 2021.

[146] 3GPP. *Support of Uncrewed Aerial Systems (UAS) connectivity, identification and tracking; Stage 2*. en. Technical Standard TS 23.256, ver. 18.1.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3853. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, June 2023.

[147] 3GPP. *Application layer support for Uncrewed Aerial System (UAS); Functional architecture and information flows*. en. Technical Standard TS 23.255, ver. 18.2.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3843. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, June 2023.

[148] 3GPP. *Study of further architecture enhancements for uncrewed aerial systems and urban air mobility*. en. Technical Report TR 23.700-58, ver. 18.1.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3993. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Mar. 2023.

[149] 3GPP. *Study on enhanced Application Architecture for UAS applications*. en. Technical Report TR 23.700-55, ver. 18.0.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3945. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Dec. 2022.

[150] 3GPP. *Study on security aspects of Uncrewed Aerial Systems (UAS)*. en. Technical Report TR 33.854, ver. 17.1.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3763. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Dec. 2021.

[151] 3GPP. *Security aspects of Uncrewed Aerial Systems (UAS)*. en. Technical Standard TS 33.256, ver. 17.3.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3930. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, June 2023.

[152] 3GPP. *Study on security of architecture enhancements for UAS, UAV and UAM*. en. Technical Report TR 33.891, ver. 18.0.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4083. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, Mar. 2023.

[153] 3GPP. *Uncrewed Aerial Systems Network Function (UAS-NF); Aerial Management Services; Stage 3*. en. Technical Standard TS 29.256, ver. 18.1.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3866. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, June 2023.

[154] 3GPP. *Uncrewed Aerial System Service Supplier (USS) Services; Stage 3*. en. Technical Standard TS 29.255, ver. 18.1.0. https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3904. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, June 2023.

[155] *5G!Drones: Unmanned Aerial Vehicle Vertical Applications' Trials Leveraging Advanced 5G Facilities*. [Website]. https://5gdrones.eu Accessed: 07 August, 2023.

[156] Lechosław Tomaszewski, Robert Kołakowski, and Sławomir Kukliński. "Integration of U-space and 5GS for UAV services". In: *2020 IFIP Networking Conference (Networking)*. https://ieeexplore.ieee.org/document/9142706. 2020, pp. 767–772.

[157] Lechosław Tomaszewski, Robert Kołakowski, and Paweł Korzec. "On 5G Support of Cross-Border UAV Operations". In: *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. 2020, pp. 1–6. DOI: 10.1109/ICCWorkshops49005.2020.9145262.

[158] *PriMO-5G: Virtual Presence in Moving Objects through 5G*. [Website]. https://primo-5g.eu/ Accessed: 07 August, 2023. URL: https://primo-5g.eu/.

[159] Lechosław Tomaszewski et al. "ETHER: Energy- and Cost-Efficient Framework for Seamless Connectivity over the Integrated Terrestrial and Non-terrestrial 6G Networks". In: *Artificial Intelligence Applications and Innovations. AIAI 2023 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Vol. 677. Cham: Springer Nature Switzerland, 2023, pp. 32–44. DOI: 10.1007/978-3-031-34171-7_2.

[160] *GOF 2.0 Integrated Urban Airspace Validation*. [Website]. https://gof2.eu/project Accessed: 07 August, 2023.

[161] GOF 2.0 consortium. *Validation of Integrated Multi-stakeholder Architecture for UAS Traffic Management: GOF 2.0 Integrated Urban Airspace*. White Paper. https://gof2.eu/wp-content/uploads/2022/12/GOF2.0-white-paper.pdf. Dec. 2022.

[162] MITRE Engenuity. *Open Generation Uncrewed Aircraft Systems (UAS) Use Cases*. Report. https://f.hubspotusercontent20.net/hubfs/7754670/Open_Generation/Open%20Gen%20Reports/ME_OpenGen_UAS_UseCases_Feb2022.pdf. Next Generation Mobile Networks, Jan. 2022.

[163] ACJA. *Interface for Data Exchange between MNOs and the UTM Ecosystem: NetworkCoverage Service Definition v1.00*. en. Technical Report ver. 1.0. Aerial Connectivity Joint Activity, Feb. 2021. URL: https://www.gsma.com/iot/wp-content/uploads/2021/02/ACJA-NetworkCoverage-Service-Definition-v1.00.pdf.

[164] Guangyi Liu et al. "Vision, requirements and network architecture of 6G mobile network beyond 2030". In: *China Communications* 17.9 (2020), pp. 92–104. DOI: 10.23919/JCC.2020.09.008.

[165] Yun Chen et al. "Pervasive intelligent endogenous 6G wireless systems: Prospects, theories and key technologies". In: *Digital Communications and Networks* 6.3 (2020), pp. 312–320. DOI: https://doi.org/10.1016/j.dcan.2020.07.002.

[166] ITU-R. *Future Technology Trends of Terrestrial International Mobile Telecommunications Systems towards 2030 and Beyond*. Report M.2516-0 (11/2022). International Telecommunication Union – Radiocommunication Sector, Nov. 2022.

[167] NGMN. *6G use cases and analysis*. White Paper ver. 1.0. Next Generation Mobile Networks (NGMN) Alliance, Feb. 2022. URL: `https://www.ngmn.org/wp-content/uploads/NGMN-6G-Use-Cases-and-Analysis.pdf`.

[168] Puneet Jain. *Rel-20 Planning and Progress in TSG SA*. 3GPP News. First published in Highlights Issue 09, December 2024. Feb. 2025. URL: `https://www.3gpp.org/news-events/3gpp-news/sa-rel20`.

[169] 3GPP. *Interface between the Control Plane and the User Plane nodes*. en. Technical Standard TS 29.244, ver. 18.5.0. 3rd Generation Partnership Project, Mar. 2024. URL: `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3111`.

[170] Jiyoung Cha et al. *RAN-CN Converged User-Plane for 6G Cellular Networks*. doi: 10.1109/GLOBECOM48099.2022.10001487. 2022.

[171] Marius Corici et al. “Organic 6G Networks : Decomplexification of Software-based Core Networks”. In: *2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. doi: 10.1109/EuCNC/6GSummit54941.2022.9815730. 2022, pp. 541–546.

[172] Abdulaziz Abdulghaffar et al. “Modeling and Evaluation of Software Defined Networking Based 5G Core Network Architecture”. In: *IEEE Access* 9 (2021). doi: 10.1109/ACCESS.2021.3049945, pp. 10179–10198.

[173] Jose Costa-Requena et al. “SDN-Based UPF for Mobile Backhaul Network Slicing”. In: *2018 European Conference on Networks and Communications (EuCNC)*. doi: 10.1109/EuCNC.2018.844279 2018, pp. 48–53.

[174] Open Networking Foundation. *Using P4 and Programmable Switches to Implement a 4G/5G UPF in Aether*. [Website]. Last accessed 10 Apr 2024. Sept. 2021. URL: `https://opennetworking.org/news-and-events/blog/using-p4-and-programmable-switches-to-implement-a-4g-5g-upf-in-aether/`.

[175] Susanna Schwarzmann et al. “An Intelligent User Plane to Support In-Network Computing in 6G Networks”. In: *ICC 2023 - IEEE International Conference on Communications*. doi: 10.1109/ICC45041.2023.10279652. 2023, pp. 1100–1105.

[176] Gyungmin Kim, Yohan Kim, and Hyuk Lim. “Deep Reinforcement Learning-Based Routing on Software-Defined Networks”. In: *IEEE Access* 10 (2022). doi: 10.1109/ACCESS.2022.3151081, pp. 18121–18133.

[177] Yingya Guo et al. *Distributed Traffic Engineering in Hybrid Software Defined Networks: A Multi-agent Reinforcement Learning Framework*. doi: 10.48550/arXiv.2307.15922. 2023. arXiv: `2307.15922 [cs.NI]`.

[178] Nicu Florin Zaicu et al. “Helix: Traffic Engineering for Multi-Controller SDN”. In: *Proceedings of the ACM SIGCOMM Symposium on SDN Research (SOSR)*. Virtual Event USA: ACM, Oct. 2021, pp. 80–87. ISBN: 978-1-4503-9084-2. DOI: 10.1145/3482898.3483354. (Visited on 08/06/2024).

[179] Lechosław Tomaszewski et al. “ETHER: Energy-and Cost-Efficient Framework for Seamless Connectivity over the Integrated Terrestrial and Non-terrestrial 6G Networks”. In: doi: 10.1007/978-3-031-34171-7_2. June 2023. ISBN: 978-3-031-34170-0.

[180] ETSI. *Network Functions Virtualisation (NFV); Management and Orchestration*. en. Group Specification, GS NFV-MAN 001 V1.1.1. https://www.etsi.org/deliver/etsi_gs/nfv-man/001_099/001/01.01.01_60/gs_nfv-man001v010101p.pdf. European Telecommunications Standards Institute, Dec. 2014.

[181] IBM. *An architectural blueprint for autonomic computing*. IBM Autonomic Computing White Paper, Fourth Edition, June 2006.

[182] NGMN. *6G Requirements and Design Considerations*. White Paper ver. 1.0. Next Generation Mobile Networks, Feb. 2023.

[183] Mårten Ericson et al. “6G Architectural Trends and Enablers”. In: *2021 IEEE 4th 5G World Forum (5GWF)*. 2021, pp. 406–411.

[184] Yanxia Xing et al. “Research on Key Technologies of 6G Network Architecture”. In: *2024 International Wireless Communications and Mobile Computing (IWCMC)*. Ayia Napa, Cyprus: IEEE, May 2024, pp. 451–455. ISBN: 9798350361261. DOI: 10.1109/IWCMC61514.2024.10592315. (Visited on 12/30/2024).

[185] Gabriele Baldoni et al. “Data-Centric Service-Based Architecture for Edge-Native 6G Network”. In: *IEEE Communications Magazine* 62.4 (Apr. 2024), pp. 32–38. ISSN: 1558-1896. DOI: 10.1109/MCOM.001.2300178. (Visited on 11/20/2024).

[186] Xueqiang Yan et al. “User-Centric Network Architecture Design for 6G Mobile Communication Systems”. In: *2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. Gothenburg, Sweden: IEEE, June 2023, pp. 305–310. ISBN: 9798350311020. DOI: 10.1109/EuCNC/6GSummit58263.2023.10188283. (Visited on 12/30/2024).

[187] Marius Corici, Eric Troudt, and Thomas Magedanz. “An Organic 6G Core Network Architecture”. In: *2022 25th Conference on Innovation in Clouds, Internet and Networks (ICIN)*. 2022, pp. 1–7.

[188] Savita Sthawarmath, Eric Renault, and Thierry Lejkin. “Designing Stateless Control-Plane for next-Generation Telecom Networks”. In: *2022 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. 2022, pp. 154–159.

[189] Gurkan Gur et al. “Integration of ICN and MEC in 5G and Beyond Networks: Mutual Benefits, Use Cases, Challenges, Standardization, and Future Research”. In: *IEEE Open Journal of the Communications Society* 3 (2022), pp. 1382–1412. ISSN: 2644-125X. DOI: 10.1109/OJCOMS.2022.3195125. (Visited on 11/20/2024).

[190] Umakant Kulkarni, Amit Sheoran, and Sonia Fahmy. “The Cost of Stateless Network Functions in 5G”. In: *Proceedings of the Symposium on Architectures for Networking and Communications Systems*. Layfette IN USA: ACM, Dec. 2021, pp. 73–79. ISBN: 978-1-4503-9168-9. DOI: 10.1145/3493425.3502749. (Visited on 09/09/2024).

[191] NGMN Alliance. *NGMN 5G White Paper*. White Paper v1.0. Next Generation Mobile Networks, Jan. 2015.

[192] 3GPP. *Management and Orchestration; Architecture Framework*. Technical Standard TS 28.533, ver. 18.1.0. 3rd Generation Partnership Project, Apr. 2024.

[193] 3GPP. *Management and orchestration; 5G performance measurements*. Technical Standard TS 28.552, ver. 17.4.0. 3rd Generation Partnership Project, Sept. 2021.

[194] 3GPP. *5G end to end Key Performance Indicators (KPI)*. en. Technical Standard TS 28.554, ver. 18.2.0. 3rd Generation Partnership Project: Sophia Antipolis, Valbonne, France, June 2023. URL: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3415.

[195] Slawomir Kuklinski and Lechoslaw Tomaszewski. “Key Performance Indicators for 5G network slicing”. In: *2019 IEEE Conference on Network Softwarization (NetSoft)*. Paris, France: IEEE, June 2019, pp. 464–471. ISBN: 978-1-5386-9376-6. DOI: 10.1109/NETSOFT.2019.8806692. (Visited on 05/17/2026).

[196] Magma (The Linux Foundation Project). *Implications of the Magma Architecture; Interoperability, Scale and Resilience*.

[197] ETSI. *Requirements and interfaces specification for management of NFV-MANO*. Group Specification ETSI GS NFV-IFA 031, v4.2.1. European Telecommunications Standards Institute, June 2021.

[198] ETSI. *Open Source MANO*. [Website]. Accessed: Feb. 22, 2022. URL: https://osm.etsi.org/.

[199] Michail-Alexandros Kourtis et al. “5G Network Slicing Enabling Edge Services”. In: *2020 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. Leganes, Spain: IEEE, Nov. 2020, pp. 155–160. ISBN: 978-1-72818-159-2. DOI: 10.1109/NFV-SDN50289.2020.9289880. (Visited on 05/17/2026).

[200] Ministerstwo Nauki i Szkolnictwa Wyższego. *Wykaz czasopism naukowych - Ministerstwo Nauki i Szkolnictwa Wyższego - Portal Gov.pl*. https://www.gov.pl/web/nauka/ujednolicony-wykaz-czasopism-naukowych. (Visited on 11/20/2025).

[201] Luis Blanco et al. “AI-Driven Framework for Scalable Management of Network Slices”. In: *IEEE Communications Magazine* 61.11 (Nov. 2023), pp. 216–222. ISSN: 1558-1896. DOI: 10.1109/MCOM.005.2300147.

[202] Lechosław Tomaszewski and Robert Kołakowski. “Mobile Services for Smart Agriculture and Forestry, Biodiversity Monitoring, and Water Management: Challenges for 5G/6G Networks”. In: *Telecom* 4.1 (Mar. 2023), pp. 67–99. ISSN: 2673-4001. DOI: 10.3390/telecom4010006. (Visited on 02/27/2025).

[203] Lechosław Tomaszewski et al. “Mobile Networks’ Support for Large-Scale UAV Services”. In: *Energies* 15.14 (Jan. 2022), p. 4974. ISSN: 1996-1073. DOI: 10.3390/en15144974.

[204] Slawomir Kukliński et al. “6G-LEGO: A framework for 6G network slices”. In: *Journal of Communications and Networks* 23.6 (Dec. 2021), pp. 442–453. ISSN: 1976-5541. DOI: 10.23919/JCN.2021.000025.

[205] Sławomir Kukliński, Robert Kołakowski, and Bartłomiej Mastej. “Migration-driven Energy Consumption Optimisation atop Cloud Continuum”. In: *2026 IEEE 12th International Conference on Network Softwarization (NetSoft)*. Berlin, Germany, June 2026. DOI: (inpress).

[206] Sławomir Kukliński, Robert Kołakowski, and Bartłomiej Mastej. “Mlops as a Service for AI-Native 6G Networks”. In: *2025 IEEE 11th International Conference on Network Softwarization (NetSoft)*. June 2025, pp. 61–66. DOI: 10.1109/NetSoft64993.2025.11080538.

[207] Lechosław Tomaszewski and Robert Kołakowski. “On the Efficient Architecture for 6G System”. In: *Artificial Intelligence Applications and Innovations. AIAI 2024 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2024, pp. 139–153. ISBN: 978-3-031-63227-3. DOI: 10.1007/978-3-031-63227-3_10.

[208] Lechosław Tomaszewski and Robert Kołakowski. “Network Slicing vs. Network Neutrality – Is Consent Possible?” In: *Artificial Intelligence Applications and Innovations. AIAI 2023 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2023, pp. 77–90. ISBN: 978-3-031-34171-7. DOI: 10.1007/978-3-031-34171-7_6.

[209] Lechosław Tomaszewski et al. “ETHER: Energy- and Cost-Efficient Framework for Seamless Connectivity over the Integrated Terrestrial and Non-terrestrial 6G Networks”. In: *Artificial Intelligence Applications and Innovations. AIAI 2023 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2023, pp. 32–44. ISBN: 978-3-031-34171-7. DOI: 10.1007/978-3-031-34171-7_2.

[210] Lechosław Tomaszewski, Robert Kołakowski, and Mirosław Zagórda. “Application of Mobile Networks (5G and Beyond) in Precision Agriculture”. In: *Artificial Intelligence Applications and Innovations. AIAI 2022 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer International Publishing, 2022, pp. 71–86. ISBN: 978-3-031-08341-9. DOI: 10.1007/978-3-031-08341-9_7.

[211] Sławomir Kukliński et al. “A Novel Architectural Approach for the Provision of Scalable and Automated Network Slice Management, in 5G and Beyond”. In: *Artificial Intelligence Applications and Innovations. AIAI 2021 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis, John Macintyre, and Lazaros Iliadis. Cham: Springer International Publishing, 2021, pp. 39–51. ISBN: 978-3-030-79157-5. DOI: 10.1007/978-3-030-79157-5_4.

[212] Lechosław Tomaszewski et al. “High Mobility 5G Services for Vertical Industries – Network Operator’s View”. In: *Artificial Intelligence Applications and Innovations. AIAI 2021 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis, John Macintyre, and Lazaros Iliadis. Cham: Springer International Publishing, 2021, pp. 71–84. ISBN: 978-3-030-79157-5. DOI: 10.1007/978-3-030-79157-5_7.

[213] Lechosław Tomaszewski, Sławomir Kukliński, and Robert Kołakowski. “A New Approach to 5G and MEC Integration”. In: *Artificial Intelligence Applications and Innovations. AIAI 2020 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis, Lazaros Iliadis, and Elias Pimenidis. Cham: Springer International Publishing, 2020, pp. 15–24. ISBN: 978-3-030-49190-1. DOI: 10.1007/978-3-030-49190-1_2.

[214] Lechosław Tomaszewski, Robert Kołakowski, and Paweł Korzec. “On 5G Support of Cross-Border UAV Operations”. In: *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. June 2020, pp. 1–6. DOI: 10.1109/ICCWorkshops49005.2020.9145262. (Visited on 02/27/2025).

[215] Lechosław Tomaszewski, Robert Kołakowski, and Sławomir Kukliński. “Integration of U-space and 5GS for UAV services”. In: *2020 IFIP Networking Conference (Networking)*. June 2020, pp. 767–772. (Visited on 02/27/2025).

[216] Sławomir Kukliński, Lechosław Tomaszewski, and Robert Kołakowski. “On O-RAN, MEC, SON and Network Slicing integration”. In: *2020 IEEE Globecom Workshops (GC Wkshps*. Dec. 2020, pp. 1–6. DOI: 10.1109/GCWkshps50303.2020.9367527.

[217] Sławomir Kukliński et al. “5G-UASP: 5G-based multi-provider UAV platform architecture”. In: *2020 6th IEEE Conference on Network Softwarization (NetSoft)*. June 2020, pp. 242–246. DOI: 10.1109/NetSoft48620.2020.9165458.

Appendix A

Publications

This chapter presents the articles that constitute the publication series forming part of the dissertation.

[A1] Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Evaluating Scalability of Hierarchical Multi-controller SDN: A Holistic Approach". In: *Journal of Network and Systems Management* 34.4 (May 2026), p. 107. ISSN: 1573-7705. DOI: `10.1007/s10922-026-10087-w` – p. 84

[A2] Robert Kołakowski, Slawomir Kukliński, and Lechosław Tomaszewski. "Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Networks". In: *2024 IFIP Networking Conference (IFIP Networking)*. June 2024, pp. 607–612. DOI: `10.23919/IFIPNetworking62109.2024.10619899` – p. 126

[A3] Robert Kołakowski et al. "Hierarchical Traffic Engineering in 3D Networks Using QoS-Aware Graph-Based Deep Reinforcement Learning". In: *Electronics* 14.5 (Mar. 2025), p. 1045. ISSN: 2079-9292. DOI: `10.3390/electronics14051045` – p. 132

[A4] Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Time-of-Day-Aware Slice Admission Control". In: *2023 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. Sept. 2023, pp. 199–204. DOI: `10.1109/MeditCom58224.2023.10266645` – p. 163

[A5] Lechosław Tomaszewski and Robert Kołakowski. "Advanced Air Mobility and Evolution of Mobile Networks". In: *Drones* 7.9 (Sept. 2023), p. 556. ISSN: 2504-446X. DOI: `10.3390/drones7090556`. (Visited on 02/27/2025) – p. 169

[A6] Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "On 6G-Enabled SDN-Based Mobile Network User Plane with DRL-Based Traffic Engineering". In: *Artificial Intelligence Applications and Innovations. AIAI 2024 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2024, pp. 124–138. ISBN: 978-3-031-63227-3. DOI: `10.1007/978-3-031-63227-3_9` – p. 193

[A7] Robert Kołakowski, Thierry Lejkin, and Victorien Romain. "Hybrid SDN-Based Data-Centric Mobile Core Network with Content-Based Routing". In: *2025 International Conference on 6G Networking (6GNet 2025)*. Paris, France, Dec. 2025, pp. 126–130. DOI: `10.1109/6GNet68413.2025.11314080` – p. 207

[A8] Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "Performance evaluation of the OSM orchestrator". In: *2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. Nov. 2021, pp. 15–20. DOI: 10.1109/NFV-SDN53031.2021.9665052 – p. 212

Evaluating Scalability of Hierarchical Multi-Controller SDN: A Holistic Approach

Robert Kołakowski[1,2*], Sławomir Kukliński[2] and Lechosław Tomaszewski[1]

[1*]Orange Innovation Poland, Orange Polska S.A., Al. Jerozolimskie 160, Warsaw, 02-326, Masovian Voivodeship, Poland.

[2]Faculty of Electronics and Information Technology, Warsaw University of Technology, Nowowiejska 15/19, Warsaw, 00-665, Masovian Voivodeship, Country.

*Corresponding author(s). E-mail(s): robert.kolakowski@orange.com; Contributing authors: slawomir.kuklinski@pw.edu.pl; lechoslaw.tomaszewski@orange.com;

Abstract

Software-Defined Networking (SDN) is envisioned to play a key role in future mobile networks due to enhanced network control flexibility and traffic engineering capabilities. Traditional centralized single-controller SDN architecture, however, faces Control Plane (CP) scalability issues that limit its effective use in large-scale carrier-grade networks. While introducing multi-controller architectures into the network design can partially alleviate the scalability problem, the estimation of benefits and costs is very complex. Moreover, state-of-the-art models are often simplified and lack real-life evaluations, making it difficult to assess the actual impact of CP distribution on the performance and deployment costs of multi-controller networks. This paper addresses these concerns by proposing theoretical models to evaluate SDN CP scalability in centralized and hierarchical SDN, considering both CP performance and consumed resources. The model is evaluated empirically using common open-source SDN emulation tools. The results show evident scalability gains in hierarchical architectures, making them suitable for carrier-grade SDN setups.

Keywords: 6G, scalability, Software-Defined Networking, distributed networks, hierarchical multi-controller SDN

1 Introduction

Software-Defined Networking (SDN) is seen as a key enabler to enhance the flexibility of 6^{th} Generation (6G) networks [1]. Traditional Internet Protocol (IP) networks use distributed routing protocols embedded in routers for making forwarding decisions. While scalable, the routing data exchange between routers, dependence on in-built protocols, and limited programmability restrict dynamic traffic control and optimization [2]. These hinder efficient 6G network implementation, particularly the areas of network resource management [3], slicing [1], virtualization [4], service function chaining [5], advanced services, such as non-terrestrial network-based ones [6].

The SDN paradigm introduces an alternative, in which, forwarding devices – Data Plane (DP) – are fully controlled by a dedicated SDN Controller (SDNC) – Control Plane (CP). In the original Centralized SDN (C-SDN) architecture, a single logically centralized SDNC having a global network view handles all network operations, e.g., traffic control, topology management, monitoring, or DP events handling, and exposes NorthBound Interfaces (NBIs) for programmatic control to dynamically adapt network behaviour. The extensive communication between SDNC and DP devices in large networks, however, can overload SDNC raising scalability concerns [7, 8]. Single SDNC also constitutes a single-point-of-failure [9] and faces placement challenges [10].

To overcome these limitations, multi-controller SDN architectures can be used [10, 11], which employ many cooperating SDNCs organized in flat [12–14] or hierarchical structures [15–17]. While CP distribution offers potential scalability gains [18–20], it introduces new issues, such as network state consistency, complexity, or controllers' coordination overhead, which aggravate with the number of controllers [21].

Although SDN CP scalability has been widely studied [18, 19, 22–25], most works rely on simulations rather than emulation-driven validation of proposed scalability models. Moreover, the existing models mainly concern generic IP networks and focus on the user traffic, overlooking the CP traffic from SDNC functions such as monitoring or topology discovery. Moreover, the models rarely quantify the impact of CP distribution. This paper addresses these gaps with the following contributions:

- We introduce mathematical formulas based on Flatto–Hahn–Wright (FHW) queueing model to describe key scalability aspects of SDN CP, i.e., CP performance and compute, storage, and connectivity resources needed to deploy CP components.
- Using an SDN emulator built on popular open-source tools, we perform measurements and empirically validate the proposed model. Based on obtained results, we demonstrate scalability improvements of distributed SDN CP and discuss the resource costs associated with deploying additional SDN controllers.

We consider SDNC working in reactive mode, where forwarding rules are computed and installed dynamically for incoming traffic. By design, the reactive path setup in SDN is the most intensive operation in CP and thus has the biggest impact on CP scalability. We focus on C-SDN and Hierarchical multi-controller SDN (H-SDN) for their global control capabilities and growing relevance to 6G architectures [26–29], omitting distributed flat architectures due to lower performance [30], high coordination overhead, and limited network visibility [20] hindering End-to-End (E2E) Quality of Service (QoS) enforcement critical in 6G. The paper is organized as follows: Section 2 reviews SDN principles. Section 3 covers related works on SDN scalability, Section 4 introduces SDN scalability model, Section 5 shows model evaluation and discusses results, Section 6 outlines future work, and Section 7 concludes the paper.

2 Software-Defined Networking

2.1 Architecture

The original Centralized SDN (C-SDN) architecture comprises three distinct planes as shown in Fig. 1 [31]:

- **Data Plane (DP)** – physical or virtual infrastructure comprised of one or more Network Elements (NEs) that have traffic forwarding capabilities (switches, routers, etc.). Each NE exposes management and control interfaces to upper-layer entities for dynamic configuration of forwarding behavior;
- **Control Plane (CP)** – comprising a logically centralized SDN Controller (SDNC) responsible for management and control of NEs. SDNC handles network monitoring data collection, topology discovery and management, and forwarding control via dynamic installation of traffic forwarding rules within DP devices. SDNC communicates with NEs over SouthBound Interface (SBI) using dedicated CP protocols e.g., OpenFlow (OF) [32]. SDNC also provides network-to-application adaptation by abstracting DP components into a logical view and exposing NBIs to upper-layer applications, which can program DP behavior;
- **Application Plane (AP)** – gathers optional and generic SDN applications that consume SDNC NBIs to adjust the network's behavior according to the application logic, e.g., provide security enhancements, intent-based management, Traffic Engineering (TE) (QoS management, load balancing, etc.), analytics, etc. Also, Application Plane (AP) can be an intermediary layer for more advanced networking mechanisms such as Network Function Virtualization or Service Chaining [33].

C-SDN relies on a single SDNC to control the entire network. While this approach provides a global network view, simplified control, and enables global optimization, it raises CP scalability concerns [34] (cf. Section 3). With the increasing network size, the number of messages exchanged between NEs and SDNC rises substantially. Considering finite SDNC service rates and CP links capacities, both CP links and SDNC are prone to overload, which can lead to performance deterioration (due to longer processing times or packet loss) or SDNC malfunctions. As SDNC constitutes a single point of failure [9, 35], the overload can result in the total network outage. Also, SDNC placement requires careful consideration [10] so as to minimize Round-Trip Time (RTT) between the controller and NEs and enable effective traffic control. To mitigate C-SDN issues, alternative architectures are often considered [34, 36]:

- **Flat** – comprising a federation of independent SDNCs managing devices in their domains. Inter-domain traffic control is achieved via cooperation of neighboring SDNCs or inter-domain protocols, e.g., Border Gateway Protocol (BGP);

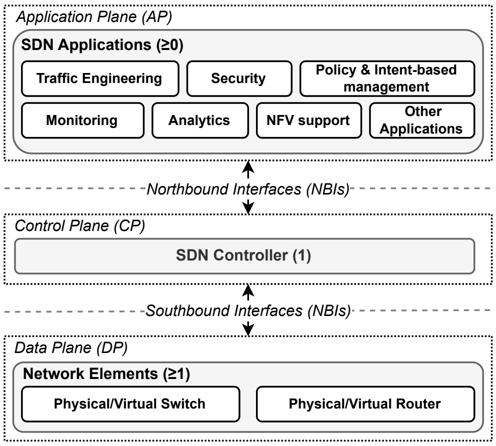


Fig. 1 Classical C-SDN architecture with example components [31].

- **Hierarchical** – comprising a federation of domain SDNCs managing devices in their domains and a centralised global controller coordinating inter-domain operations;
- **Hybrid** – integrating both SDN devices managed by SDNCs (centralized or distributed CP) and non-SDN equipment running legacy routing protocols, e.g., Open Shortest Path First (OSPF) or BGP.

Multi-controller deployments enhance fault tolerance and scalability of the network and mitigate the controller placement problem caused by the geographic distribution of CP devices [35]. However, the complexity of E2E routing and QoS control is increased, as cooperation of controllers is required to compute appropriate domain (local) and inter-domain paths and assign the flow. The latter necessitates installation of forwarding rules at the domains' border nodes (gateways), which requires extensive data exchange between SDNCs that scales with the number of deployed controllers. Moreover, maintaining consistent network state information across controllers is challenging and requires frequent exchange, leading to overhead [24, 35]. Other drawbacks are the lack of a global network view in the case of a flat architecture, or increased complexity and latency overhead due to multi-level control in the case of H-SDN. Hybrid SDN improves scalability by reducing the number of managed SDN-enabled NEs. This results in lower flexibility of traffic and E2E QoS control, and unpredictable behavior in non-SDN segments [36] – both making hybrid SDN difficult to use in carrier-grade mobile networks [7, 8].

2.2 Routing and path setup

Routing functionality is typically embedded in SDNC [32], which computes network paths and enforces them by installing forwarding rules in NEs. Three rule installation strategies can be used by SDNC[37, 38]:

- **Proactive** – paths are set before traffic arrival;
- **Reactive** – paths are set dynamically at runtime upon traffic arrival (one-shot or hop-by-hop; cf. Fig. 2);
- **Hybrid** – combining the proactive and reactive mode.

SDNC sets up the paths using dedicated CP protocols, most notably OF [32, 39]. OF introduces forwarding control via traffic abstraction called flows – the sequences of packets sharing common attributes (source/target IP/Medium Access Control (MAC) address, Differentiated Services Code Point (DSCP), ingress port, etc.). OF switches match incoming packets to flows defined by SDNC and forward them to output ports according to associated instructions. The flow abstraction enables logical traffic groupings and fine-grained routing control, which contribute to CP scalability.

Fig. 2 shows the reactive E2E path setup in OF-based C-SDN and H-SDN networks. Path setup is the most CP-intensive operation. When the traffic arrives at the switch's ingress port, the switch parses packet headers and looks up matching forwarding rules based on header metadata (source/destination IP/MAC, DSCP, etc.). If no match is found, the packet is encapsulated in an OF *packet_in* message (a1) and sent to SDNC via CP channel (a2). SDNC queues and processes the message (a3) calculates the path for the flow (a4), and responds with *flow_mod* messages (a5), containing rules to be installed by SDN switches (a6).

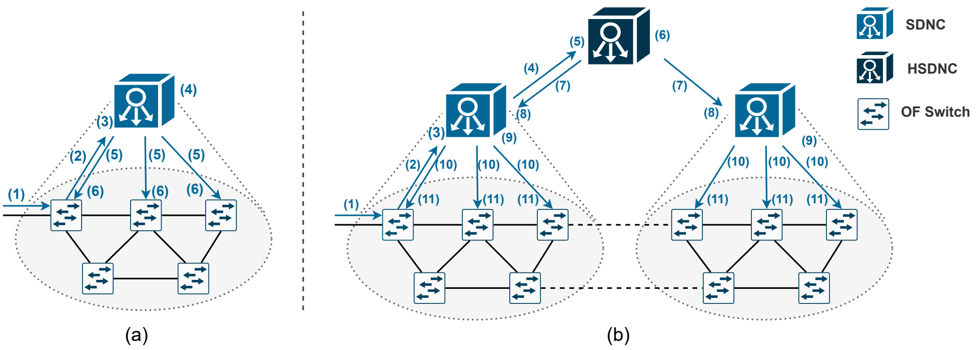


Fig. 2 Reactive path setup process in OF-based SDN networks: C-SDN (a) and H-SDN (b).

In H-SDN, the intra-domain routing process is the same as in C-SDN. For inter-domain flows (i.e., if the flow's target is in another domain), SDNC forwards *packet_in* message to the centralised Hierarchical SDN Controller (HSDNC) (b4), where it is queued (b5). Then, HSDNC computes an E2E path (b6) using an abstracted topology composed of domain gateways, inter-domain links and abstracted intra-domain links. Then, it splits the E2E path by domains into segments and instructs domain SDNCs to enforce each segment (b7). SDNCs queue these requests (b8), compute intra-domain paths (b9), and propagate *flow_mod* messages to switches (b10) for forwarding rules installation (b11). Thus, HSDNC coordinates only inter-domain flow setup, while domain SDNCs manages local routing.

3 Related work on SDN scalability

Scalability of the SDN CP is widely recognized as a key limitation of SDN [22, 23, 25, 30, 40]. Despite numerous proposed architectures and mechanisms, the scalability of SDN CP remains a multi-dimensional issue that requires careful evaluation during the network design phase [21] and as such represents a fundamental concern in production SDN-based systems. Broadly, the problem can be alleviated through three main strategies:

1. adopting distributed CP architectures that split workloads across multiple controllers [12–14, 16–18, 20, 23–25, 34];
2. improving controller design and implementation, e.g., through parallel processing and optimized control applications [22, 41–45]
3. enhancing NEs and CP protocols to reduce CP exchange and offload SDN controllers [41, 46–49].

Additionally, SDN CP scalability can be influenced by higher-level network management and orchestration frameworks through efficient controller coordination, optimized monitoring data collection, automation, and system-level load balancing, among other measures. In the following subsections, we review the key works on SDN CP scalability along three complementary dimensions: controller architectures and controller-level extensions, SDN-based network management systems, and scalability measurement and modelling techniques.

3.1 SDN controllers

Over the years, multiple SDN controller solutions have been proposed, which address the CP scalability issue in different ways [50]. The early controllers, such as NOX [51], POX[52], Floodlight [53], followed a single-controller C-SDN design, mainly serving as extensible platforms for SDN experimentation. Later C-SDN controller solutions introduced mechanisms to enhance CP scalability, e.g., Maestro adopts multi-threaded event processing to boost CP throughput [42], OpenDaylight [43] employs multiple queues for parallel event processing, ONOS [44] combines logical centralization with physical distribution for SDNC instances scaling, and Ryu [45] supports process-level parallelism. To address the CP scalability issues by design, several new SDN frameworks with distributed CP have emerged. Prominent examples include flat multi-controller architectures (cf. Section 2.1) like ONIX [12], DISCO [13], or ELASTICON [14], or hierarchical ones such as Orion [16], or Kandoo [17]. Recently, considerable effort has been focused on the development of the Teraflow controller [15] driven by ETSI, which evolved from a single-controller C-SDN architecture to a versatile one supporting both flat and hierarchical distributed deployments and integrating with popular orchestrators. The SDNC implementation itself can introduce CP scalability bottlenecks. In [54], a data-driven dependability assessment framework, DASON, is proposed to detect issues in SDNC software implementations related to protocols, scalability, performance, availability, and operational aspects. The analysis of ODL and ONOS controllers revealed over 500 issues, including delayed heartbeats, performance degradation under high load, misbehavior during flow installation, and state synchronization problems in multi-controller setups – each negatively impacting the scalability of SDN CP.

While these approaches contribute to CP scalability, the exact gains are a subject of research. For the C-SDN variant, vertical scalability, i.e., how performance scales with compute resources assigned to the SDNC instance, has been analysed in [23]. Three principles were identified for scalable SDN system design: maintenance of Control and Data Plane separation, limiting the impact of CP on DP, and minimizing the impact of network monitoring data collection on CP performance.

The CP scalability in terms of manageability and functional scalability has been analysed in [22]. The authors claim that scalability is an inherent design challenge for both SDN and non-SDN networks and propose solutions to improve SDN CP scalability, e.g., SDNC discovery mechanism to enhance resilience and combining proactive and reactive path setup (cf. Section 2.2) to optimize CP load. The gains of SDN-driven routing compared to legacy protocols such as OSPF have been shown in [55].

In [35], the key challenges faced by multi-controller SDN frameworks are described, including network state information consistency across controllers, inter-controller exchange overheads for state synchronization and E2E routing operations, or interoperability. A method to maintain a consistent network view across SDNCs has been proposed in [24]. The authors introduce a Coordinate Controller acting as a unified network operating system, abstracting multi-domain resources and provisioning E2E services. The analysis of CP overhead in distributed flat SDN architectures during routing has been described in [20]. While CP distribution generally improved the performance for simple routing algorithms, more advanced ones requiring more frequent inter-controller exchange or supporting QoS control, featured significant deterioration of scalability with the increasing number of SDNCs.

In reactive operation mode (cf. Section 2.2), SDN CP scalability is impacted by both CP protocol design and capabilities of DP devices (packet parsing, flow table lookups, OF-message creation, etc.). The key limitation comes from the high volume of CP traffic generated by incoming flows [41], further amplified by monitoring data collection [56–58] and topology discovery via OpenFlow Discovery Protocol (OFDP) [59] [60], all carried over CP channel and processed by SDNC. As OF lacks native QoS support, QoS extensions (e.g., for latency measurements) will further aggravate this issue. Moreover, OF switches suffer from limited flow table capacity due to storage and processing costs, thus requiring effective flow rules placement [38]. A potential solution to these issues is designing CP protocols from scratch by using Programming Protocol-independent Packet Processors (P4) language [46]. While many P4-based mechanisms contributing to CP scalability have been designed, e.g., TE algorithms [47, 61], In-band Network Telemetry [48], or QoS routing schemes [49], the commercial adoption is limited and declining following the discontinuation of the key P4-based NIC, Tofino.

Another way to partially mitigate CP overload and improve scalability is to dynamically reassign switches across SDNCs to reduce load imbalance across controllers. In [62], the authors propose an elite genetic algorithm to address the unbalanced SDNCs load caused by the static mapping of switches to SDNCs. The proposed switch migration algorithm demonstrates efficient load distribution and enhanced convergence compared to baseline methods, making it a good fit for evolving network environments. Similarly, [63] presents a dynamic controller-to-switch assignment approach that employs the Multi-Agent Deep Deterministic Policy Gradient algorithm to optimize load and CP latency for controllers deployed in Low Earth Orbit (LEO) satellite networks.

The scalability of multi-controller architectures is also influenced by the performance and placement of individual controllers and resulting bottlenecks. In [64], a reliability-focused controller placement approach is introduced, based on the NSGA-II algorithm, which optimizes placement of multiple SDNCs, while considering key CP link parameters such as CP path reliability and length, inter-controller links reliability, or CP load distribution. The delay-aware and energy-efficient controller placement problem is addressed in [65]. The proposed network model incorporates both power consumption and delay constraints and determines optimal controller placement using a binary integer programming approach. Since most power consumption originates from the baseline operation of switches and controllers, energy-efficient placement involves reducing the number of controllers, which can adversely affect overall CP performance. Furthermore, a controller placement strategy for hierarchical multi-orbit satellite networks is proposed in [66]. The genetic algorithm optimizes the placement of domain SDNCs on the ground and in LEO orbits, as well as upper-layer controllers on Medium Earth Orbit (MEO) nodes, aiming to minimize the delay to set up forwarding rules and reduce synchronization delays across hierarchical controllers to improve network state consistency. The latter issue has been extensively studied in [67], where an SDNC placement strategy that optimizes state synchronization under failure-free and single-link-failure scenarios is proposed. In [68], the authors emphasize that controller placement should be optimized jointly with flow request distribution across controllers, since controller performance is highly sensitive to even small increases in request arrival rates. The formulated controller placement and scheduling problem (CPSP) is solved by adopting a greedy algorithm that forwards bursting requests in one sub-network to selected controllers in neighboring sub-networks to handle CP load variations.

3.2 SDN-based network management systems

In modern telecommunication systems, SDN controllers perform both network management (Fault, Configuration, Accounting, Performance, Security (FCAPS)) and control functions such as routing, path computation, topology discovery, or traffic engineering. For today, the performance of management, including KPIs, is neither standardized nor well addressed in the literature. While the vast majority of traffic handled by the SDN controller comes from CP functions, the impact of management operations on scalability cannot be ignored.

The most prominent example is the implementation of monitoring over CP channel in traditional OF-based networks. Several strategies are adopted to solve this issue. In [69], two heuristic algorithms for optimizing flow monitoring in single and multi-controller networks are presented. The algorithms minimize monitoring cost by selecting reporting switches and/or receiving controllers based on criteria such as reporting delay, monitoring cost, or number of assigned flows. In [23], the authors propose to deploy a statistics server, external to the controller (but synchronized with it), that performs OF-based monitoring via dedicated channels and exposes metrics on the controller's demand. In [70], a WAN SDN Controller (WSC) architecture is introduced, which automates TE and improves operational efficiency of enterprise 5G backhaul networks. To reduce overhead associated with metric collection, WSC incorporates a Telemetry Collector that enables decentralized metric subscription, fine-grained metric analysis, and selective storage and reporting, thereby minimizing monitoring traffic. The WSC field trials conducted in an enterprise 5G network operated by Chunghwa Telecom in Taiwan demonstrated up to 80% reduction in monitoring traffic, substantially enhancing network performance and scalability.

Moreover, the management and orchestration frameworks that run on top of SDN can further impact scalability in CP. In [71], a QoS-based WAN Infrastructure Manager (Q-WIM) built on the ETSI MANO framework and leveraging SDN-based transport is proposed. The framework supports the deployment, management, and optimization of virtualized network functions and network slices across geographically distributed environments, i.e., NFVI-PoPs interconnected via an SDN-based transport network. Q-WIM enables the enforcement of QoS policies (minimum/maximum burst size, maximum bitrate, minimum/maximum bandwidth, and marked, tranmission and congestion delays). An implementation based on the ONOS controller demonstrated fast path setup times below 1 s for medium-sized networks (fewer than 100 switches). Performance, however, degraded rapidly beyond this scale, indicating CP scalability issues. The hierarchical orchestration architecture for multiple OpenFlow-based SDN controllers, called OrchFlow, is proposed in [72]. The framework comprises a centralized orchestrator that acts as middleware between applications requesting network services and heterogeneous controllers managing different subdomains. The centralized orchestrator coordinates SDN controllers via OrchFlow Local Agents (OLAs), which expose a unified RESTful interface for controlling SDN resources. The architecture was evaluated using Ryu and Floodlight controllers with TCP and UDP traffic, under proactive, reactive, and hybrid flow setups. Although scalability is not explicitly analyzed, the results show that flow setup times increase with domain count in reactive and hybrid modes, indicating coordination overhead.

An inconsistent network state across SDN controllers can lead to network faults that require corrective actions by the SDN controller, thereby increasing CP load [73, 74]. A Distributed, Elastic, and Fault-tolerant State Management of Network Function, DEFT, is proposed in [73]. While the framework focuses on Network Function (NF) state management, SDN plays a crucial role in rerouting traffic to NF instances in the event of failure or overload, and in orchestrating state synchronization between NF instances. In [74], the orchestrator agent for event synchronization across multiple SDN controllers is proposed. The method relies on simultaneously modulating incongruous flows (via synchronized *packet_in* generation) and synchronizing *flow_mod* events to avoid redundant *packet_in* messages during path updates. The method enables better distribution of flow requests in time, improving the performance and scalability of the SDN controller.

Another way to improve scalability in SDN network management systems is by enhancing CP resilience [75–77] or parallel operations. In [75], a custom Programmable

Data Plane-based Controller Protection Protocol (PCPP) is introduced, which effectively prevents CP overload by spoofed or malformed DP requests. A concept for resilient SDN controller is proposed in [76]. The framework employs a master controller with multiple backups that verify the master's flow rules to mitigate the impact of faults or a compromised master controller. The ParaCon architecture [77] features a parallel CP that distributes path computation across multiple controllers sharing a global network view to compute multiple smaller path segments. Another

In [78], an SDN architecture for handling a hybrid data plane for cloud-native systems is proposed. The system integrates with Kubernetes and introduces segment routing over IPv6 (SRv6) for DP forwarding. SRv6 behavior is fully configurable by the SDN controller using OF protocol. The evaluation demonstrated efficient SRv6 policy updates, supporting the elastic scaling of microservice instances. SDN can also be efficiently applied in mobile networks. In [29], the authors discuss the benefits of using SDN for handover management in 4G networks and substitution of GPRS Tunneling Protocol (GTP) by tunnel-less IP routing. The concept can be smoothly extended and applied to both 5G and 6G networks.

SDN scalability has been widely studied for SDN control functions, yet it is rarely investigated in the context of SDN-based network management frameworks. This gap motivates our work to develop a scalability model that encompasses both control and management traffic to reliably predict CP performance in SDN-based systems.

3.3 Scalability measurement and modeling

SDN scalability measurement usually considers performance metrics related to isoefficiency scalability (ability to maintain parallel efficiency across processors with the increasing problem and system size) and isospeed scalability (ability to maintain execution speed with the increasing problem and system size) [25]. As these metrics might not reflect the actual performance of SDN CP, the alternative scalability metric is proposed and tested for various SDN CP designs [25]. The impact of communication latency of networked control systems, which can be generalized to the SDN environment, has been thoroughly analyzed and modeled [18]. The results show that distributed control architectures can offer improved performance compared to the centralized ones if all-to-all exchange is considered (as in the classical C-SDN approach). Moreover, a diminished size of domains reduces the algorithmic delay induced by path computation operations [79, 80], further improving scalability.

The key metrics related to SDN CP scalability are the performance of SDNC in terms of path creation and flow assignment, and SDNCs deployment costs. A comprehensive analysis of 34 different SDNs controllers and benchmarking tools, with a detailed evaluation of the nine most popular SDNC implementations, has been presented [81]. Although limited to single-controller deployments, it provides insights into achievable performance thresholds. Another benchmark of 6 open-source SDNCs, which followed practices standardized by Internet Engineering Task Force (IETF), has been presented in [82]. The study focuses on performance (message processing time and rate, reactive path provisioning, topology changes detection) and scalability (expressed by topology discovery time and size), highlighting the impact of single- vs. multi-threaded SDNC implementation [82]. Existing benchmarking tools, however, such as Cbench [83], usually focus on testing SDNCs responsiveness (typically OF handshake messages) – which is not a reliable performance indicator in the context of an operational environment, where multiple message types are generated with different types and processing logic. Moreover, SDNC performance can fluctuate due to the presence of other assisting services, e.g., routing, monitoring, or security functions or hardware platforms. The packet processing delay in Open vSwitch (OVS) – the most common OF virtual switch – was evaluated and modeled [84]. Results show that processing does not induce delays exceeding $\approx 60\,\mu s$ and is characterized by relatively small delay jitter, making OVS a good DP emulation choice for benchmarking SDNC solutions.

A performance model for evaluation of the response time and scalability of centralized, decentralized, and hierarchical architectures has been proposed [19], which proves better scalability of hierarchical and flat multi-controller SDN over C-SDN. The costs of deploying such architectures are not considered, however. The comprehensive model of SDN CP scalability has been proposed in [30]. The theoretical evaluations have shown that a hierarchical model outperforms centralized and distributed flat ones in terms of scalability. While comprehensive, the model neglects the impact of topology discovery protocols and monitoring suite on SDNC service rate, which is a commonly indicated problem [23]. The Artificial Intelligence (AI)-driven performance-prediction model for SDN-enabled data center networks has been proposed in [85]. While the work focuses on optimizing DP configuration to accommodate heterogeneous workloads (elephant and incast traffic), the performance predictions are tightly coupled with CP performance and can therefore serve as input for CP optimization.

Although widely studied, the existing SDN CP scalability models focus either on CP performance or deployment costs, making direct comparisons of architecture variants problematic. This paper bridges the gap by introducing a comprehensive model for evaluating CP performance and scalability of C-SDN and H-SDN.

4 SDN scalability model

SDN CP scalability refers to the system's ability to sustain operational efficiency under increasing flow arrival rates. It is mainly determined by the CP performance, which in turn depends on:

- SDN CP architecture – single vs. multi-controller, with or without logical centralization;
- SDNC features – compute capacity, queuing mechanism (parallelisation, prioritisation, etc.), and routing algorithm;
- Network topology graph features – size (number of nodes, edges, and hosts), density, degree distribution, etc.;
- SDNC queue load – varying with flow arrivals and SDNC control traffic (e.g., topology management, monitoring).

The proposed approach focuses on modeling logically centralized C-SDN and logically distributed H-SDN (cf. Fig. 2). We model SDN network as a graph $G(V, E)$, where V – network vertices (switches) and E – network edges. The network flows are initiated and consumed by hosts H that are attached to G via dedicated host links E_H. In the multi-controller case, we define the unique domain identifiers as $\mathfrak{d}$ belonging to the domain set $\mathfrak{d} \in \mathfrak{D}$. Each network domain is managed by its own SDNC deployed in close proximity to switches. Each domain topology is represented by a graph $G^{\mathfrak{d}}(V^{\mathfrak{d}}, E^{\mathfrak{d}})$ and connected with neighboring domains via inter-domain links that terminate at domain gateways – switches or access points for hosts. For simplification, we consider the balanced network in which each domain has approximately the same number of attached hosts and nodes. E2E operations are conducted by HSDNC, which operates on an overlay network view $G^{\psi}(V^{\psi}, E^{\psi})$ consisting of gateways, inter-domain links, and abstracted intra-domain links representing connectivity between a pair of domain gateways. In analyses, we consider L2 switches fully controlled by SDNCs (as in OF-based networks). To evaluate the scalability of SDN CP, we consider two commonly considered factors [21, 23, 25, 35, 40]:

- **CP performance** – measured with: i) controllers' utilization for given CP load; and ii) Time-To-Serve (TTS), which is the time required by SDNC to dynamically create a path and assign flow to it (install forwarding rules in switches).
- **Resource consumption** – compute, storage and connectivity resources consumed by CP components deployed in a specific CP architecture.

While not exhaustive, the above metrics enable easy interpretation and comparison of different SDN architecture variants. To evaluate scalability, we introduce the CP scalability index Γ described in Section 5.5.

4.1 Control Plane Performance

4.1.1 SDN CP utilization:

We model the SDNC/HSDNC as an M/M/1 queue [86], where λ – flow request arrival rate from individual hosts and R – controller's service rate denoting the number of CP messages processed per time unit. A key factor related to CP scalability is SDNC utilization U:

$$U = \frac{\lambda}{R}; \quad U \geq 0 \tag{1}$$

The SDN CP should maintain the conditions in which the SDNC service rate exceeds the flow request rate ($U < 1$). In the opposite case ($U \geq 1$), the CP event queue length will grow indefinitely, leading to destabilization and the collapse of the system. Operating at high U also contributes to Time-To-Serve (TTS) due to longer waiting time for events in the SDNC event queue.

We model the flow arrival using the Poisson process. The total flow arrival rate in the SDN system depends on the number of hosts H and their frequency of transmission. Each host can issue up to $(|H| - 1)$ flow requests to remaining hosts with mean arrival rate λ. Considering that both flow inter-arrival and service times are exponentially distributed, following [30], we can model SDNCs and HSDNC as M/M/1 queues. Additionally, each controller has to handle CP traffic ζ that is generated by SDN components regardless of user traffic. These refer to traffic generated by SDN always-on protocols and mechanisms such as topology discovery or monitoring (cf.

Table 1). The load contributed by ζ typically increases linearly with the number of network nodes and hosts and can vary over time. Also, we consider the local scope of these operations, i.e., that domain messages are not propagated to other controllers.

Table 1 SDN CP traffic types and examples

Symbol	Type	Traffic examples
λ	flow arrival rate	TCP/UDP flows issued by hosts
$\zeta_c, \zeta_{\mathfrak{d}}$	CP event rate generated by protocols at SDNC level for C-SDN (ζ_c) and H-SDN ($\zeta_{\mathfrak{d}}$)	Topology discovery protocols (e.g., LLDP, OFDP), DP monitoring (e.g., OF-based metrics pulling), SDN events (e.g., port status, keepalive events)
ζ_ψ	CP event rate generated by SDNCs (only for H-SDN)	Abstracted domain topology updates, Domain monitoring (domain metric aggregates), SDNC events (e.g., inter-domain flow entry expiration notification, domain status messages, alarms)

For C-SDN, the total CP messages arrival rate and SDNC utilization is given by:

$$\lambda_c = \lambda|H|(|H|-1) + \zeta_c(H,V,t) \tag{2}$$

$$U_c = \frac{\lambda_c}{R_c} \tag{3}$$

$$\zeta_c(H,V,t) = \begin{cases} f(H,V,t), & \text{if domain\_id} = c \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

In H-SDN, computing CP event rate and utilization is more complex, as only a subset of SDNCs participates in the path setup process. While intra-domain flows are handled by SDNCs, inter-domain flow requests are first sent to and processed by HSDNC, and then, delegated to relevant SDNCs for execution (see Section 2.2 and Fig. 2 for details). To model this behavior, we use a fork-join queuing structure (similar to FHW model [87]) in which SDNCs and HSDNC form a chain of M/M/1 queues, and transition between queues is driven by stochastic parameters of the traffic (cf. Fig. 3)

The flow request rate λ_c, comprising both intra- and inter-domain flows, is split across $|\mathfrak{D}|$ domains and handled by SDNCs:

$$\lambda_c = \sum_{\mathfrak{d}}^{|\mathfrak{D}|} \lambda_c \eta_{\mathfrak{d}} \tag{5}$$

where $\sum_{\mathfrak{d}}^{|\mathfrak{D}|} \eta_{\mathfrak{d}} = 1$ and $\eta_{\mathfrak{d}} \in (0,1)$ represents a fraction of flow requests handled by the controller of domain $\mathfrak{d}$. Each flow is an inter-domain one with probability α, meaning that each SDNC forwards $\alpha\lambda_c\eta_{\mathfrak{d}}$ requests to be processed by HSDNC, yielding the total HSDNC queue filling rate of $\alpha\lambda_c$. To set up an inter-domain path, HSDNC

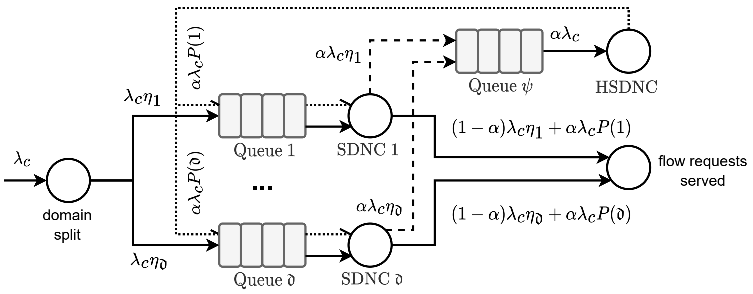


Fig. 3 Flow requests queuing model for H-SDN CP variant.

instructs a number of domain SDNCs. We consider that SDNC selection depends on two factors – the number of domains n traversed by the inter-domain flow and whether the domain belongs to the flow path. Two probability functions can then be formulated $P(L = n)$ – the probability of path traversing n domains (L being the random variable representing path length in domains), and $P_\mathfrak{d}(n)$ – the probability that domain $\mathfrak{d}$ is a part of the path. The overall probability that domain $\mathfrak{d}$ controller is selected is hence given by:

$$P(\mathfrak{d}) = \sum_{k}^{|\mathfrak{D}|} P(L = k) P_\mathfrak{d}(k) \tag{6}$$

Each SDNC must handle additional inter-domain flow requests that fill its queue at a rate of $\alpha\lambda_c P(\mathfrak{d})$, and operational overhead $\zeta_\mathfrak{d}(H^\mathfrak{d}, V^\mathfrak{d}, t)$. Therefore, the total request rate $\lambda_\mathfrak{d}$ handled by a single SDNC in H-SDN is:

$$\lambda_\mathfrak{d} = \lambda_c\eta_\mathfrak{d} + \alpha\lambda_c P(\mathfrak{d}) + \zeta_\mathfrak{d}(H^\mathfrak{d}, V^\mathfrak{d}, t) \tag{7}$$

The load of HSDNC is determined by the inter-domain flow rate $\alpha\lambda_c$, and operational overhead $\zeta_\psi(\mathfrak{D}, t)$, which scales with the number of domains, giving the total HSDNC event rate λ_ψ:

$$\lambda_\psi = \alpha\lambda_c + \zeta_\psi(\mathfrak{D}, t) \tag{8}$$

For service rates of SDNC $R_\mathfrak{d}$ and HSDNC R_ψ, controller utilizations are given by:

$$U_\psi = \frac{\lambda_\psi}{R_\psi} = \frac{\alpha\lambda_c + \zeta_\psi(\mathfrak{D}, t)}{R_\psi} \tag{9}$$

$$U_\mathfrak{d} = \frac{\lambda_\mathfrak{d}}{R_\mathfrak{d}} = \frac{\lambda_c\eta_\mathfrak{d} + \alpha\lambda_c P(\mathfrak{d}) + \zeta_\mathfrak{d}(H^\mathfrak{d}, V^\mathfrak{d}, t)}{R_\mathfrak{d}} \tag{10}$$

$$\lambda_c = \lambda|H|(|H| - 1) \tag{11}$$

In the simplest case of equally-sized domains $\eta_{\mathfrak{d}} = 1/|\mathfrak{D}|$, and uniform path lengths $L \sim \mathcal{U}\{2, 3, ..., |\mathfrak{D}|\}$, with equal selection probability for each domain controller:

$$P(L = n) = \frac{1}{|\mathfrak{D}| - 1}, \quad P_{\mathfrak{d}}(n) = \frac{n}{|\mathfrak{D}|} \tag{12}$$

Then:

$$P(\mathfrak{d}) = \sum_{k}^{|\mathfrak{D}|} P(L = k) P_{\mathfrak{d}}(k) = \frac{1}{|\mathfrak{D}|(|\mathfrak{D}| - 1)} \sum_{k=2}^{|\mathfrak{D}|} k = \frac{|\mathfrak{D}| + 2}{2|\mathfrak{D}|}$$

and the domain controller utilization can be simplified to:

$$\lambda_{\mathfrak{d}} = \frac{\lambda_c}{|\mathfrak{D}|} + \alpha\lambda_c \frac{|\mathfrak{D}| + 2}{2|\mathfrak{D}|} + \zeta_{\mathfrak{d}}(H^{\mathfrak{d}}, V^{\mathfrak{d}}, t) \tag{13}$$

$$U_{\mathfrak{d}} = \frac{\frac{\lambda_c}{|\mathfrak{D}|} + \alpha\lambda_c \frac{|\mathfrak{D}|+2}{2|\mathfrak{D}|} + \zeta_{\mathfrak{d}}(H^{\mathfrak{d}}, V^{\mathfrak{d}}, t)}{R_{\mathfrak{d}}} \approx \frac{\frac{\lambda_c}{|\mathfrak{D}|}(1 + \frac{\alpha(|\mathfrak{D}|+2)}{2}) + \frac{\zeta_c(H,V,t)}{|\mathfrak{D}|}}{R_{\mathfrak{d}}} \tag{14}$$

It is worth noting that the probability of controller selection and α coefficient (Eq. 6) depends on multiple features such as network topology, domain type (e.g., edge, cloud), split of switches across controllers, or individual flow features (influenced by service type and corresponding logic, endpoints location, etc.).

4.1.2 Time-To-Serve:

Time-To-Serve (TTS) is a key SDN CP performance metric, which describes the time needed by CP components to set up a path for the flow requests. In the real-life system, TTS is affected by several factors, including:

- RTT between switches and controller and between SDNC and HSDNC (in H-SDN);
- algorithmic delay of the routing algorithm, dependent on the network graph size;
- queuing and processing delays in DP devices (packet buffering, packet headers parsing, matching, *packet_in* generation, or installation of rules);
- queuing and processing delays in CP components (event processing logic, state updates, etc.).

To formulate TTS for C-SDN and H-SDN, we follow the procedures and notation introduced in Sections 2.2 and 4.1.1 (see Table 2). The time duration of the path setup step x is denoted as τ_x^y, where $y \in \{a, b, ab\}$ indicate C-SDN, H-SDN or both CP variants. The worst-case TTS can be computed using eq. 15–4.1.2, where τ_{TTS} – TTS in C-SDN, and $\tau_{TTS}^{\psi}, \tau_{TTS}^{\mathfrak{d}}$ – TTS for inter- and intra-domain flows in H-SDN.

$$\tau_{TTS} = \max_{v \in V} (\tau_1^a(v) + \tau_2^a(v)) + \tau_3^a + \tau_4^a(V, E) + \max_{v \in V} (\tau_5^a(v) + \tau_6^a(v)) \tag{15}$$

15

$$\tau_{TTS}^{\mathfrak{d}} = \tau_{TTS}; \quad \text{for:} \quad V = V^{\mathfrak{d}}, E = E^{\mathfrak{d}} \tag{16}$$

$$\begin{aligned}\tau_{TTS}^{\psi} = & \max_{\mathfrak{d}\in\mathfrak{D}, v\in V^{\mathfrak{d}}} \left[\tau_1^b(v) + \tau_2^b(v) + \tau_3^b(\mathfrak{d}) + \tau_4^b(\mathfrak{d})\right] + \tau_5^b + \tau_6^b(V^{\psi}, E^{\psi}) \\ & + \max_{\mathfrak{d}\in\mathfrak{D}, v\in V^{\mathfrak{d}}} [\tau_7^b(\mathfrak{d}) + \tau_8^b(\mathfrak{d}) + \tau_9^b(V^{\mathfrak{d}}, E^{\mathfrak{d}}) + \tau_{10}^b(v) + \tau_{11}^b(v)]\end{aligned} \tag{17}$$

Table 2 CP performance model parameters, units and value ranges (unitless parameters marked as "–")

Symbol	Quantity	Unit	Range
V	network nodes (switches)	–	$\mathbb{N}$
E	network edges (links)	–	$\mathbb{N}$
H	network hosts	–	$\mathbb{N}_0$
λ	flow requests arrival rate	$1/s$	$\mathbb{R}_0^+$
R	controller service rate	$1/s$	$\mathbb{R}^+$
U	controller utilization	–	$\mathbb{R}_0^+$
α	probability of inter-domain flows	–	$[0, 1]$
ζ	operational overhead	$1/s$	$\mathbb{R}_0^+$
n	domains traversed by inter-domain flow	–	$\mathbb{N}$
P	probability of SDNC selection during inter-domain path setup	–	[0,1]
$\lvert\mathfrak{D}\rvert$	number of SDN domains	–	$\mathbb{N}$
τ_x	duration of step x in reactive path setup	s	$\mathbb{R}^+$
τ_{TTS}	Time-To-Serve (TTS)	s	$\mathbb{R}^+$
β	computation capabilities factor	s	$\mathbb{R}^+$

For CP evaluation scenarios, the formulas can be simplified. Packet parsing, matching, and rule installation delays are often negligible, even for software switches, i.e., $\tau_1^{ab} \leq 40\,\mu s$ s [84], $\tau_6^a, \tau_{11}^b \sim \mu s$. Propagation delays over fiber (τ_2^a, τ_5^a, $\tau_2^b, \tau_4^b, \tau_7^b, \tau_{10}^b$) are most often constant and symmetric for uplink (UL) and downlink (DL) CP channel (i.e., $\tau_2^a \approx \tau_5^a$, $\tau_2^b \approx \tau_{10}^b$, etc.). Moreover, in H-SDN, SDNC is typically deployed close to DP components to minimise RTT. For silica fibers (refractive index ≈ 1.44), propagation delay is $4.81\,\mu s/km$ yielding RTT $\leq 1\,ms$ for Metro SDN Networks (round-trip distance up to 100 km). With smaller domain and placement of SDNC closer to switches, $\tau_2^b, \tau_{10}^b \sim \mu s$.

The delays induced by SDNCs/HSDNC depend on the used routing algorithm and processing and queuing latency. In the simple case of Dijkstra Shortest-Path (SP) routing with Fibonacci heap implementation, the time complexity is $O(E \log V)$. Delay is computed using constants $\beta_c, \beta_{\mathfrak{d}}, \beta_{\psi}$, representing computation capabilities of SDNCs and HSDNC, and capturing hardware resources and software efficiency. Since β can be measured empirically, path calculation times can be estimated:

$$\tau_4^a \approx \beta_c E \log V$$

$$\tau_6^b \approx \beta_\psi E^\psi \log V^\psi$$
$$\tau_9^b \approx \beta_\mathfrak{d} E^\mathfrak{d} \log V^\mathfrak{d} \tag{18}$$

Compared to C-SDN, H-SDN greatly reduces intra-domain routing delay. For an even network graph split ($V_\mathfrak{d} = \frac{V}{|\mathfrak{D}|}, E_\mathfrak{d} = \frac{E}{|\mathfrak{D}|}$), the gain can be computed as follows (cf. Fig. 4):

$$\frac{\tau_9^b}{\tau_4^a} = \frac{\beta_\mathfrak{d} E^\mathfrak{d} \log V^\mathfrak{d}}{\beta_c E \log V} = \frac{\frac{E}{|\mathfrak{D}|} \log \frac{V}{|\mathfrak{D}|}}{E \log V} = \frac{1}{|\mathfrak{D}|}\left(1 - \frac{\log |\mathfrak{D}|}{\log V}\right) \tag{19}$$

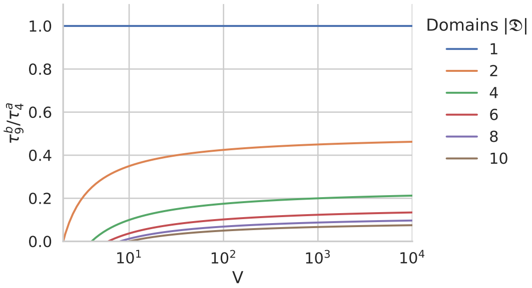


Fig. 4 Algorithmic delay gains for different numbers of domains.

The last term that contributes to TTS is related to processing and queuing delay and can be estimated following M/M/1 queuing model and Little's Law[88]:

$$\tau = \frac{1}{R - \lambda}; \quad \lambda < R \tag{20}$$

Hence, the delays induced by controllers and final simplified TTS formulas are:

$$\tau_3^a = \frac{1}{R_c - \lambda_c}$$
$$\tau_3^b = \tau_8^b = \frac{1}{R_\mathfrak{d} - \lambda_\mathfrak{d}} \tag{21}$$
$$\tau_5^b = \frac{1}{R_\psi - \lambda_\psi}$$

$$\tau_{TTS} \approx 2 \cdot \max_{v \in V} \tau_2^a(v) + \frac{1}{R_c - \lambda_c} + \beta_c E \log V \tag{22}$$

$$\tau_{TTS}^\mathfrak{d}(V^\mathfrak{d}, E^\mathfrak{d}) = \tau_{TTS}(V^\mathfrak{d}, E^\mathfrak{d}) \tag{23}$$

$$\tau^{\psi}_{TTS} \approx 2 \cdot \max_{\mathfrak{d} \in \mathfrak{D}, v \in V^{\mathfrak{d}}} \left[\tau^b_2(v) + \tau^b_3(\mathfrak{d}) + \tau^b_4(\mathfrak{d})\right] + \tau^b_5 + \tau^b_6(V^{\psi}, E^{\psi}) + \tau^b_9(V^{\mathfrak{d}}, E^{\mathfrak{d}})$$
$$= 2 \cdot \max_{\mathfrak{d} \in \mathfrak{D}, v \in V^{\mathfrak{d}}} \left[\tau^b_2(v) + \frac{1}{R_{\mathfrak{d}} - \lambda_{\mathfrak{d}}} + \tau^b_4(\mathfrak{d})\right] + \frac{1}{R_{\psi} - \lambda_{\psi}} + \beta_{\psi} E^{\psi} \log V^{\psi} + \beta_c E^{\mathfrak{d}} \log V^{\mathfrak{d}} \tag{24}$$

It must be noted that TTS depends on topology $(V, E, H, \mathfrak{D})$, flow requests arrival rate λ, but also on specific configuration of controllers $(\zeta, \beta, R$, etc.).

4.2 Resource consumption

Deploying distributed SDN CP introduces additional compute, storage, and connectivity resources to support multiple controller instances and inter-controller communication. Accurate resource estimation is challenging due to hardware–software variability and parallel processing. Hence, we propose a data-driven method to project resource requirements for both C-SDN and H-SDN (see Table 3 for model parameters summary). For each controller instance, we consider a resource vector $X = [C, D, M, B]^T$ denoting total consumption of: compute, disk I/O, memory, and CP channel bandwidth. This resource consumption is driven by:

- static consumption X_{stat} – software footprint X_0, and resources used by always-on controller mechanisms operating regardless of user traffic load (cf. Table 1) – denoted by $X_{prot}(U_{prot})$,
- dynamic consumption X_d – caused by processing of incoming network flows by controllers.

The above can be expressed by eq. 25 (1):

$$X(U) = X_{stat} + X_{dyn}(U) = X_0 + X_{prot}(U_{prot}) + X_{dyn}(U, U_{prot}) \tag{25}$$

As resource consumption scales differently with SDNC load across resource types, we define $f(U)$ to map controller utilization to actual resource usage:

$$f(U, U_{prot}) = \begin{bmatrix} k_C U e^{\gamma_C U} \\ k_D U \\ k_M U e^{\gamma_M U} \\ max(k_B U, B_m) \end{bmatrix} \tag{26}$$

The function $f(U)$ models SDN software behaviour under increasing CP load and crossing capacity thresholds. We adopt linear scaling for CP bandwidth (limited by CP link capacity B_m), linear scaling for disk I/O operations (logging, network state data retention, etc.), and exponential scaling for CPU and memory. For the latter, a non-linear model is used to capture resource spikes beyond the controller's service rate R. Coefficients k_C, k_D, k_M, k_B represent usage per utilisation unit and γ_M, γ_C are overload penalty factors. These constants are specific to the controller's implementation and deployment and can be derived via controller profiling in static operation (i.e., under no user traffic) and regression (cf. Section 5). The dynamic consumption increase associated with user traffic is hence equal to $X_{dyn} = f(U) - f(U_{proto})$ for $U \in \langle U_{proto}, 1\rangle$. Based on the above, in C-SDN, the total resource consumption of a single SDNC at utilisation U_c, can be expressed by:

$$X^{(c)} = X_0^{(c)} + X_{prot}^{(c)}(U_{prot}) + f^{(c)}(U_c) - f^{(c)}(U_{proto}) = \begin{bmatrix} C_{stat}^{(c)} \\ D_{stat}^{(c)} \\ M_{stat}^{(c)} \\ B_{stat}^{(c)} \end{bmatrix} + \begin{bmatrix} C_{dyn}^{(c)} \\ D_{dyn}^{(c)} \\ M_{dyn}^{(c)} \\ B_{dyn}^{(c)} \end{bmatrix} \tag{27}$$

$$= \begin{bmatrix} C_0^{(c)} \\ D_0^{(c)} \\ M_0^{(c)} \\ B_0^{(c)} \end{bmatrix} + \begin{bmatrix} C_{prot}^{(c)}) \\ D_{prot}^{(c)} \\ M_{prot}^{(c)} \\ B_{prot}^{(c)} \end{bmatrix} + \begin{bmatrix} k_C^{(c)}(U_c e^{\gamma^{(c)} U_c} - U_{prot} e^{\gamma^{(c)} U_{prot}}) \\ k_D^{(c)}(U_c - U_{prot}) \\ k_M^{(c)}(U_c e^{\gamma^{(c)} U_c} - U_{prot} e^{\gamma^{(c)} U_{prot}}) \\ \min[k_B^{(c)} U_c, B_m] - k_B^{(c)} U_{prot}) \end{bmatrix}$$

For H-SDN, total resource consumption $X^{(h)}$ is a sum of SDNCs $X^{(\mathfrak{d})}$ and HSDNC $X^{(\psi)}$ consumptions, where $U_{\mathfrak{d}}, U\psi$ are their utilisations (cf. Section 4.1.1):

$$X^{(h)} = X^{(\psi)} + \sum_{\mathfrak{d} \in \mathfrak{D}} X^{(\mathfrak{d})}; \quad \text{where:} \tag{28}$$

$$X^{(\mathfrak{d})} = X_0^{(\mathfrak{d})} + X_{prot}^{(\mathfrak{d})}(U_{prot}^{(\mathfrak{d})}) + f^{(\mathfrak{d})}(U_{\mathfrak{d}}) - f^{(\mathfrak{d})}(U_{prot}^{(\mathfrak{d})})$$

$$X^{(\psi)} = X_0^{(\psi)} + X_{prot}^{(\psi)}(U_{prot}^{(\psi)}) + f^{(\psi)}(U_{\psi}) - f^{(\psi)}(U_{prot}^{(\psi)})$$

Table 3 Resource model parameters, units, and value ranges (unitless parameters marked as "–").

Symbol	Quantity	Unit	Range
U	controller utilization	–	$\mathbb{R}_0^+$
X	total resource consumption	–	–
X_{stat}	static resource consumption	–	–
X_d	dynamic resource consumption	–	–
C	CPU consumption	%	$\mathbb{R}^+$
D	disk I/O consumption	$\frac{\text{MiB}}{\text{s}}$	$\mathbb{R}^+$
M	memory consumption	MiB	$\mathbb{R}^+$
B	bandwidth consumption	Mbps	$\mathbb{R}^+$
k_C	CPU utilization coefficient	%	$\mathbb{R}^+$
k_D	disk I/O utilization coefficient	$\frac{\text{MiB}}{\text{s}}$	$\mathbb{R}^+$
k_M	memory utilization coefficient	MiB	$\mathbb{R}^+$
k_B	bandwidth utilization coefficient	Mbps	$\mathbb{R}^+$
γ_x	CP saturation penalty factor	–	$\mathbb{R}^+$

5 Evaluation

This section presents the foundational evaluation of the proposed SDN CP performance model using an emulation environment. First, we present the evaluation methodology and its impact on metrics, measurement testbed, and the conducted test scenarios. Then, the measurement results are compared with the models' predictions to validate the proposed model. Finally, we discuss the impact of SDN CP distribution on scalability, along with key challenges of CP design in large-scale SDN networks.

5.1 Testing assumptions

We assume approximately equal-sized domains with the same number of nodes and hosts, and average node degree $deg(V^{\mathfrak{d}}) = deg(V)$. Workloads are evenly distributed across domain controllers ($\sum_{\mathfrak{d}\in\mathfrak{D}} X^{(\mathfrak{d})} \approx |\mathfrak{D}|X^{(\mathfrak{d})}$). The implementations of SDNCs and HSDNC are the same across tests ($R_c = R_{\mathfrak{d}}$, $R\psi > R_c$), including the used Dijkstra routing algorithm ($\beta_\psi \approx \beta_c \approx \beta$).

For easier comparison, HSDNC and SDNC in C-SDN are deployed in the same location (i.e., $\tau_2^a + \tau_5^a \approx 2\tau_2^b + 2\tau_4^b = \tau_{rtt}$), with identical switch placement across both architectures. In H-SDN, SDNCs are deployed in close proximity to the switches ($\tau_2^b, \tau_{10}^b \approx 0$). Finally, HSDNC sees only a fraction of the domain topology in the overlay graph, i.e., $\psi_r : \psi_r V = V^\psi$. The above assumptions allow us to simplify TTS formulas to:

$$\tau_{TTS} \approx \tau_{rtt} + \frac{1}{R - \lambda_c} + \beta \cdot deg(V) \cdot V \log V \tag{29}$$

$$\tau_{TTS}^{\mathfrak{d}} \approx \frac{1}{R - \lambda_{\mathfrak{d}}} + \beta \cdot deg(V) \frac{V}{|\mathfrak{D}|} \log \frac{V}{|\mathfrak{D}|} \tag{30}$$

$$\tau_{TTS}^{\psi} \approx \tau_{rtt} + \frac{2}{R - \lambda_{\mathfrak{d}}} + \frac{1}{R - \lambda_\psi} + \beta \cdot deg(V) \left(\psi_r V \cdot \log \psi_r V + \frac{V}{|\mathfrak{D}|} \log \frac{V}{|\mathfrak{D}|} \right) \tag{31}$$

Finally, τ_{RTT} is a constant component for TTS and dependent on spatial distances between the controller and DP. To focus on the flow processing dynamics, we consider a small metro SDN network, in which RTT becomes negligible ($\tau_{RTT} << 1\,ms$).

5.2 Methodology

The conducted evaluation SDN CP had three key goals:

- assess the impact of topology and CP load on SDN CP performance metrics (utilisation, TTS, and resource consumption),
- validate proposed models and assess their usability in SDN performance estimation and CP design,
- evaluate scalability tradeoffs for C-SDN and H-SDN CP architectures.

The tests are conducted in synthetic mesh topologies (20 and 40 nodes, average node degrees 3 and 4). To each topology, we attach 5–50 hosts (with a step of 5) that act as flow sources and sinks. To provide enough headroom in the DP, each link had 1 Gbps capacity. Each host generates mice Transport Control Protocol (TCP) flows with arrival time following a Poisson distribution and arrival rates corresponding to moderate and high load (new flow every 15 or 30 s, i.e., $\lambda = 0.066$ or $\lambda = 0.033$ flows/s). The actual flow arrival rate seen by the controller is $\lambda_x = \lambda * |H|(|H| - 1)$ (cf. Section 4.1.1). Each test lasts 120 s to capture the behaviour of CP under various load conditions, namely static, dynamic, non-overloaded, and overloaded state at CP saturation. In the tests, we focus on TCP flows due to in-built rate control and retransmissions, allowing us to track TTS increase close to overload states, when packets are dropped by the controller or switches (the latter due to the in-switch buffer retention). Finally, we test H-SDN with the network split into 2 and 4 domains.

For model validation, we measured parameters and coefficients introduced in performance metrics formulas (cf. Section 4) and listed respective values in Table 4.

Table 4 Constants computed using single-test data for resource consumption forecasting.

Const.	**Unit**	**C-SDN**	**H-SDN**	
		SDNC (c)	**SDNC** ($\mathfrak{d}$)	**HSDNC** (ψ)
β	s		$4.28 \cdot 10^{-8}$	
R	$1/s$	140		850
γ_C	–	1.19		10^{-16}
γ_M	–	2.38		3.86×10^{-9}
U_{prot}	%	22.8	$\approx 22.8/\lvert\mathfrak{D}\rvert$	0.03
C_{stat}	%	0.018	$\approx 0.018/\lvert\mathfrak{D}\rvert$	0.017
k_C		3.15		4.23×10^{-17}
D_{stat}	MiB/s	7.65×10^{-4}	$\approx 7.65/\lvert\mathfrak{D}\rvert \times 10^{-4}$	1.74×10^{-3}
k_D		0.09		1.48
M_{stat}	MiB	60.42	$\approx 60.42/\lvert\mathfrak{D}\rvert$	35.74
k_M		2.55		1.93×10^{-10}
B_{stat}	$Mbps$	2.32×10^{-4}	$\approx 2.32 \times 10^{-4}/\lvert\mathfrak{D}\rvert$	8.72×10^{-4}
k_B		156.29		27774.26
ψ_r	–	–	–	$\psi_r = 0.5$

To calculate β, we recorded path computation times on random Erdős Renyi graphs (up to 100 nodes, varying node degrees), divided them by Dijkstra's algorithm time complexity, and averaged the results. Resource-specific coefficients were derived by measuring idle and total resource usage (X_{stat} and X), calculating $X_d = X - X_{stat}$, and performing regression to obtain respective coefficient values (linear regression for B, D, and least squares method for C, M). To profile U_{prot}, we measured the average utilization of controllers during the idle phase of operation (with no user traffic). For SDNCs, this included bursty traffic from LLDP-based topology discovery, ARP host discovery and OF-based monitoring. To simplify the analysis, we excluded acquisition of flow related metrics that can quickly overload SDNC (the issue further discussed

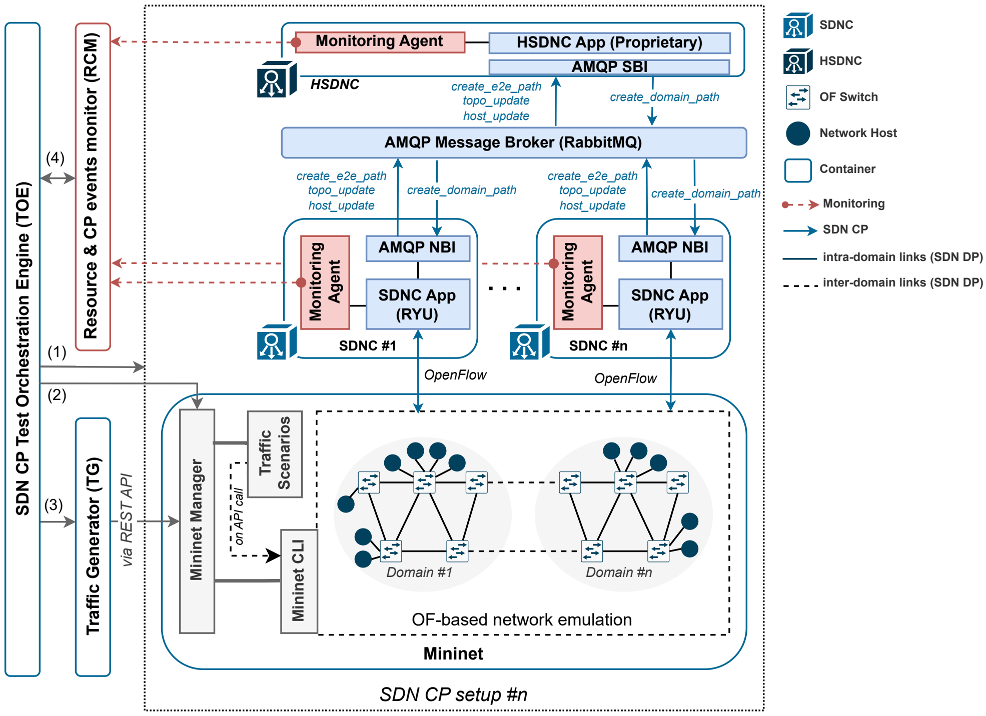


Fig. 5 Evaluation testbed and test orchestration steps.

in Section 5.5). For HSDNC, U_{prot} results from SDNCs updates of abstracted domain topology and related metrics. The measurements were conducted for a single topology and setup and reused across test scenarios. Controller service rates were measured via stress testing to identify the max service rate before severe performance degradation.

5.3 Testbed

The tests were conducted in the Python-based SDN testbed shown (cf. Fig. 5). The testbed comprises two main parts:

- tested SDN CP setup – a complete emulation of the SDN environment comprising an OF-based network, domain SDNCs, HSDNC, and a message broker for CP messages exchange between global and local controllers.
- Test orchestration and measurement suite – Traffic Generator (TG), Resource & SDN CP events monitor (RCM), and SDN CP Test Orchestration Engine (TOE)

Each testbed component is containerized using Docker [89] and connected using the workstations' host network to reduce latency from routing between virtual interfaces and obtain more realistic test conditions. Containers had no resource limits to allow unrestricted scaling.

To implement the SDN network, we used the latest version of Mininet [90] (2.3.0) and OVS switches [91] for DP emulation. The Mininet container comprises a dedicated Manager that acts as an adapter, enabling configuration and interactions with the Mininet environment via the Mininet CLI.

SDNC are implemented on the basis of Ryu controller [45]. Each SDNC within its main application loop (Ryu App) implements basic operations related to routing (shortest path), topology discovery (ARP), handling *packet_in*, *flow_mod*, and flow expiration events, as well as supports configurable metrics acquisition over the CP channel (port and flows statistics and metrics processing). HSDNC is fully implemented in Python; however, the main application loop (HSDNC App) mimics the Ryu App's behaviour (a single queue for network events and handler-based processing). Each controller implements an AMQP interface to connect with the RabbitMQ message broker [92] to exchange CP messages. HSDNC communicates with SDNCs using a custom OF-like protocol that implements four basic operations:

- *create_e2e_path* – issued by SDNCs on *packet_in* events for inter-domain flows (step b4 Fig. 2). The message contains the originating domain, source/destination IP, source/destination MAC addresses, and the switch ID that issued OF *packet_in* event;
- *create_domain_path* – sent by HSDNC to instruct SDNCs to create path segments in domains for inter-domain flow (step b7 Fig. 2). The message contains the source and destination switches' IDs, and ingress/egress ports (required to enforce the path between neighboring domains);
- *topo_update* and *host_update* – sent by SDNCs to update HSDNC about domain topology changes). These messages contain, respectively, serialized NetworkX graphs [93] and lists of hosts connected to each SDN domain, along with key metadata (ID, IP, MAC address, etc.).

To retain the source and destination of each CP message, an individual topic has been created for the controller that contains the SDN domain ID. For upstream queue messages (i.e., *create_e2e_path* sent by SDNC to HSDNC), SDNC includes the originating domain ID in the CP message. Finally, to measure CP performance metrics, each SDNC App interacts with the dedicated Python-based Monitoring Agent. Its goal is twofold. First, it records all CP events received by the controller and enhances them with contextual information (event type, timestamp, protocol, size, flow ID, etc.). Second, using *psutil* Python library, it samples resource usage by the controller processes at a 100 ms interval. Collected data is continuously flushed to RCM.

Each test (120 in total) was orchestrated by TOE. The test orchestration process comprises 4 steps:

1. Instantiation of Docker containers [89] using docker compose. The test configuration is stored in a dedicated YAML file that contains the test duration, the Mininet Topology configuration, the number of domains, and DP traffic parameters.
2. Once all containers are up and running, TOE uploads the Mininet Topology configuration via the REST API exposed by Mininet Manager. Then, the latter

uses the Mininet CLI to start up the OF-based network emulation using the uploaded Mininet Topology configuration.

3. Then, TOE uploads the traffic parameters and topology to TG. TG generates the traffic scenario (JSON list comprising flow ID, source/destination IP, bandwidth, duration, and offset). The traffic scenario is later uploaded to the Mininet container and started via the Mininet Manager API. Based on the configuration, Mininet Manager launches client/server iPerf3 [94] sessions in Mininet hosts via Mininet CLI. In each test, every flow had a duration of 2 s and a throughput of 1 kbps.
4. The last step involves fetching all data collected by RCM and storing it locally for further analysis. Once the data is fetched, TOE tears down the containers using the *docker compose down* command to perform a full cleanup before launching the next scenario.

The testbed was run on a workstation with an Intel Core i7-12700K 12$^{\text{th}}$ Gen, 5.0 GHz with 12 Central Processing Unit (CPU) cores, 32 GB Random Access Memory (RAM), and 512 GB NVMe 4.0 Solid-State Drive (SSD) storage drive.

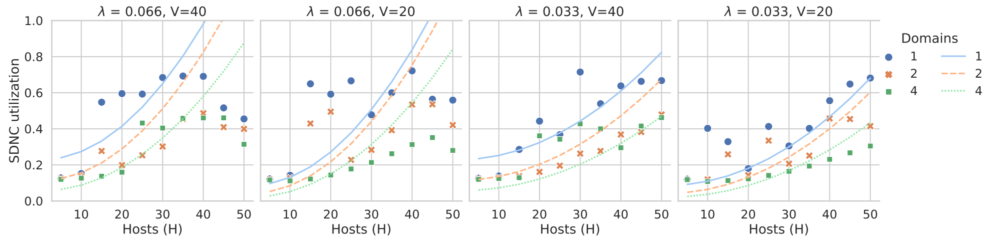
5.4 Measurement results and validation

5.4.1 SDN CP performance metrics

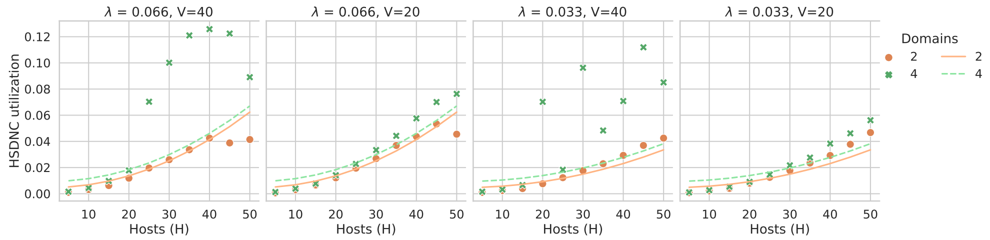
Fig. 6 presents the theoretical and measured SDN CP performance metrics for C-SDN and H-SDN – average controller utilization, TTS for intra- and inter-domain flow setups, and flow rejection rate.

First, the observed controller utilization stays below 80%, even when overloaded ($\lambda = 0.066$, $|H| \geq 45$). Controllers' queues saturation leads to the collapse of performance and dropping *packet_in* messages (reflected in increased flow rejection rate). As a result, SDNC registers a lower number of *packet_in* events, causing under-reported utilization. Moreover, HSDNC utilization for 4 domains exceeds model-driven boundaries at high loads due to long TTS for inter-domain flows. Delayed rule installation can trigger TCP-driven retransmissions and generation of multiple *packet_in* and *create_e2e_path* events for a single flow. Remaining utilization inconsistencies on a domain level can also be caused by short-term CP load peaks and overload (e.g., due to periodic topology discovery) that result in an $\approx 10\%$ flow rejection rate (cf. Fig. 6e).

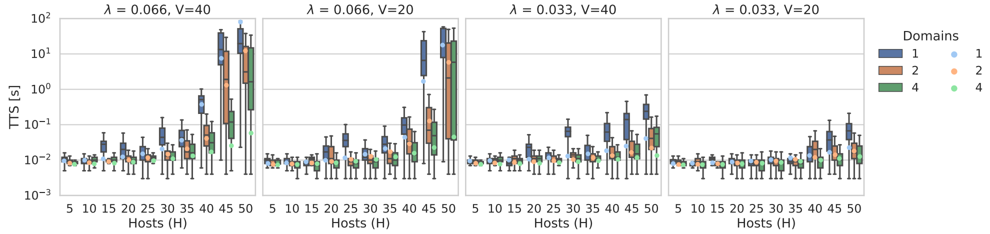
TTS grows hyperbolically with the number of hosts for both C-SDN and H-SDN. Control distribution across multiple controllers reduces per-controller utilisation, due to the split of path setup messages and static CPload over multiple controller instances. Moreover, for increasing network load, the increase of both TTS and Inter-Quartile Range (IQR) of TTS can be observed. While IQR in H-SDN case is much bigger, TTS jitter actually indicates better responsiveness of distributed CP, as a large portion of flows are handled with much lower delay compared to C-SDN. For high loads in H-SDN, the 3$^{\text{rd}}$ quartile of intra-domain flows TTS is consecutively below the 1st quartile of C-SDN. Also, TTS explosion occurs earlier for 2 than 4 domains,



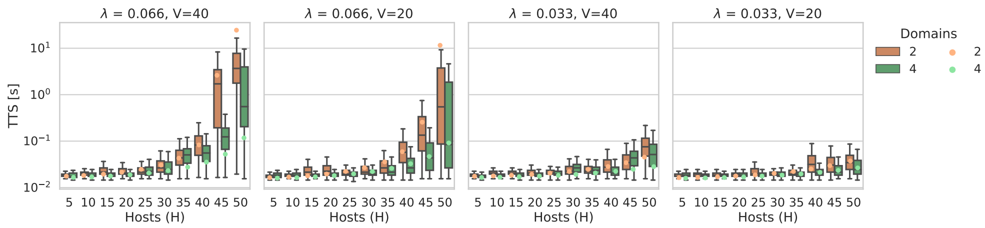
(a) Utilization of SDNCs (dot – measured, line – model).



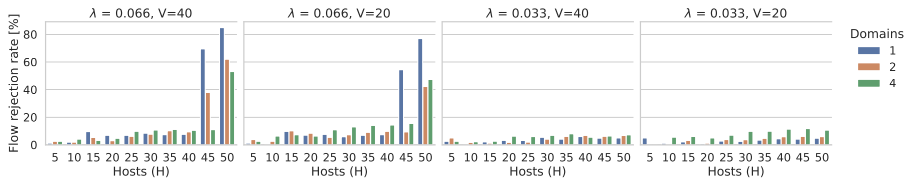
(b) Utilization of HSDNC (dot – measured, line – model).



(c) TTS: intra-domain flow setup (box – measured, dot – model).



(d) TTS: inter-domain flow setup (box – measured, dot – model).

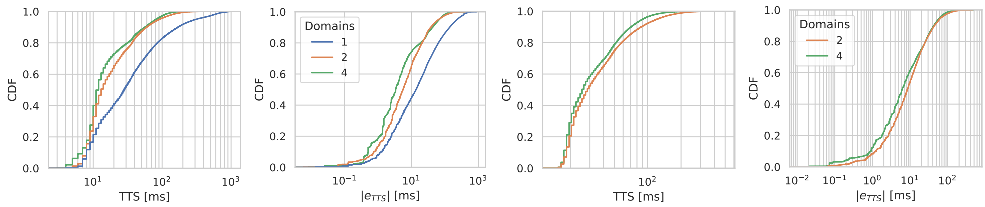


(e) Total flow rejection rate.

Fig. 6 C-SDN and H-SDN performance for various flow rates λ, domains $|\mathfrak{D}|$, nodes V and hosts H.

especially for the larger topology. Finally, a shorter delay can be observed for smaller topologies due to reduced static controller load.

To validate TTS formulae, we computed theoretical values using equations given in Section 4 and compared them with measurement data to obtain residuals and Cumulative Distribution Functions (CDFs) (cf. Fig. 7). Model prediction accuracy is further evaluated using Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R^2, in relation to measured TTS statistics – mean μ and standard deviation σ (cf. Table 5). The validation is conducted jointly for C-SDN and H-SDN.



(a) Intra-domain TTS (b) Intra-domain TTS residuals (c) Inter-domain TTS (d) Inter-domain TTS residuals

Fig. 7 CDF for TTS values and residuals for flow setup.

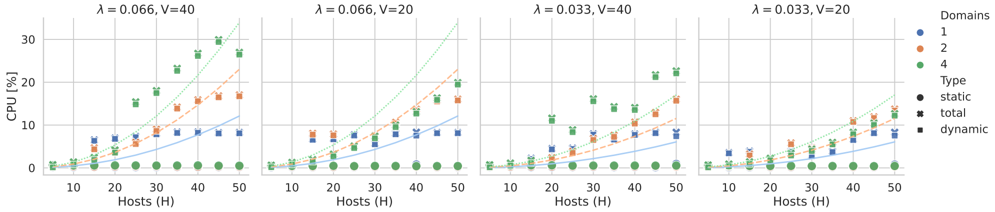
First, for a given number of hosts and flow arrival rates, TTS for intra-domain flows exhibits very high variance – typical for a dynamic queue-based system under high load. In this case, the model explains ≈ 80% of the variance with $MAE \approx 10$ ms. The error, however, is heavily influenced by TTS values in the near-overload conditions in the C-SDN scenario. In multi-domain setups (cf. Fig. 7a), the errors are lower than 10 ms for 60% (2 domains) and 80% (4 domains) of samples. For inter-domain flows, slightly lower accuracy stems from the varied performance of underlying controllers, leading to larger latency variance (both relay time to HSDNC and execution of HSDNC instructions). Still, the model explains over 60% of variance with MAE lower than 5.5 ms across all tests and network loads. In both cases, even in the presence of outliers, the model features good explanatory capabilities considering the noisiness of TTS in SDN networks. Overall, the model performs better for larger network topologies, due to the smaller impact of local anomalies in the network on the SDN workflows.

Table 5 Model accuracy metrics for TTS for intra- and inter-domain flows.

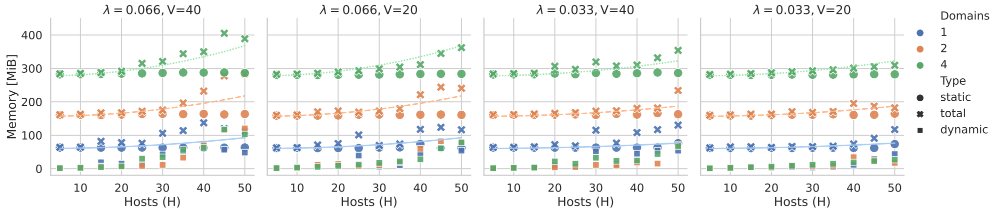
Metric	**Intra-domain flows**		**Inter-domain flows**	
	outliers	**w/o outliers**	**outliers**	**w/o outliers**
RMSE [ms]	29.45	21.28	9.98	7.67
MAE [ms]	11.66	10.17	6.32	5.38
R^2	0.68	0.80	0.57	0.61
$\mu[ms]$	27.24	25.75	29.51	28.53
$\sigma[ms]$	51.96	47.28	15.36	12.40

5.4.2 Resource utilization

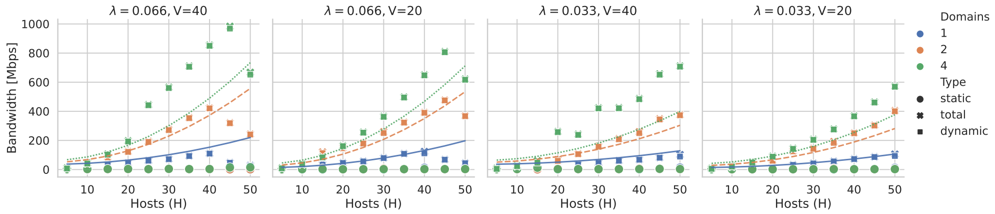
In a virtualized environment, dynamic resource consumption of containers depends on running software, host OS, virtualization technology, and resource allocation mechanisms, typically leading to high burstiness and jitter. Therefore, we analyze only the time-averaged resource consumption of CP components for each test scenario and CP setups. Fig. 8 shows the measured static X_{stat}, dynamic X_{dyn}, and total consumption X, and model-driven approximations for different topologies and flow arrival rates.



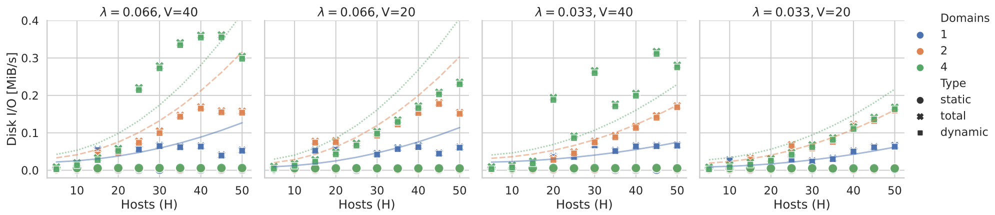
(a) CPU consumption (C)



(b) RAM consumption (M)



(c) CP bandwidth (B)



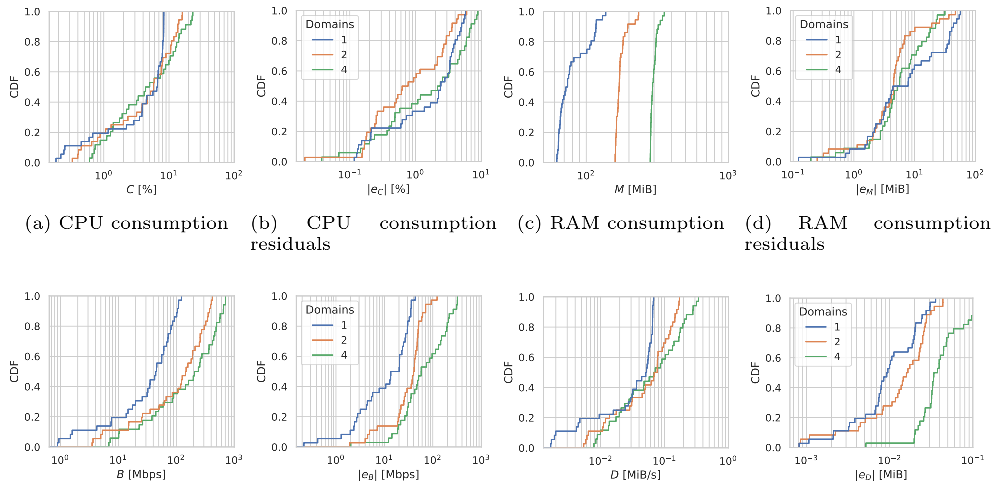
(d) Disk I/O bandwith (D)

Fig. 8 Average resources consumed by SDN CP components.

First, static consumption of C_{stat}, D_{stat}, and B_{stat} is comparable across different CP setups and almost negligible, except for M_{stat}, which rises nearly linearly with the number of controllers. The proposed function $f(U, U_{prot})$ grasps the dynamic consumption trends of individual resources: exponential for M_{dyn}, and linear for D_{dyn} and B_{dyn}. For C_{dyn}, the exponential coefficients are very low, making the scaling almost linear (cf. Fig. 8). This is likely due to the tests conducted on the high-performance workstation, which had free resources, scaling, and allocation for the controllers. Deploying on less powerful virtualized infrastructure (such as telco clouds) would likely produce more significant non-linear coefficients.

It can be observed that dynamic CPU consumption is highly volatile for high network loads, likely due to uneven CP operations split across domain SDNC (higher individual utilization due to exponential growth that dominates CPU consumption reported by other SDNCs). In terms of M_{dyn}, the actual usage matches the model almost perfectly with only minor deviations in the C-SDN case. Both D_{dyn} and B_{dyn} rise approximately linearly with the number of hosts. The measured values, however, largely vary across tests, which might be caused by fluctuations of the number of CP events emitted by DP components, also impacting the number of disk-related operations (e.g., logging). It must be noted, however, that for large CP loads, the controller's rejection rate significantly rises (cf. Fig. 6e), resulting in a noticeable change of trend.

To validate the resource consumption model, we follow the approach described in Section 5.4.1. The results are shown in Fig. 9 and Table 6.



(a) CPU consumption (b) CPU consumption residuals (c) RAM consumption (d) RAM consumption residuals

(e) Bandwidth (f) Bandwidth residuals (g) Disk I/O (h) Disk I/O residuals

Fig. 9 Cumulative distribution functions for resource consumption measured values and residuals.

The adopted resource scaling model explains a large share of variance ($R^2 \in \langle 0.65, 0.96 \rangle$), with very low error in terms of the projection of CPU and RAM consumption. Higher discrepancies in measured and model-driven values of bandwidth and disk I/O (reflected in high MAE) are linked to multiple outliers for both B_{dyn} and D_{dyn} (especially in the case of four domains, cf. Fig. 8). Finally, while in the experiments we aimed to maintain a balanced split of topology and control across SDNCs, achieving exactly the same load per-domain for quasi-random graphs was not always possible and contributed to estimation errors. Nonetheless, the proposed model enables accurate estimation of resources required by the SDN CP components, especially considering the high noise conditions inherent to SDN networks.

Table 6 Resource model accuracy metrics for different CP setups.

Metric	**Resource type**			
	CPU [%]	**RAM [MiB]**	**Bandwidth [Mbps]**	**Disk I/O [MiB/s]**
RMSE	3.29	17.65	89.69	0.04
MAE	2.42	11.48	57.26	0.03
R^2	0.65	0.96	0.76	0.67
μ	7.53	193.07	191.63	0.09
σ	6.42	95.77	215.98	0.08

5.5 Scalability and CP design challenges

To evaluate the scalability of SDN CP architectures, we use the scalability index Γ:

$$\Gamma = \prod_{l \in L} \left(\frac{M_c^l}{M_h^l} \right)^{1/L} \tag{32}$$

where M_c^l, M_h^l are metric values measured respectively for C-SDN (baseline) and H-SDN variant for test $l \in L$. The index uses geometric mean to mitigate the distortion induced by outliers and provide a typical percentage gain from using a certain SDN CP architecture. Table 7 shows Γ computed for both performance and resource metrics for two extreme cases (small vs. large topology and load). For τ^{ψ}_{TTS}, we use C-SDN's $\tau^{\mathfrak{d}}_{TTS}$ as the baseline, as it corresponds to E2E path setup time for a given network topology. The evaluation of U considers only domain controllers. For resource parameters, we consider the total resource consumption (X) of the CP variant.

Table 7 Scalability Index Γ for H-SDN with two and four domains.

Scenario	$\lvert\mathfrak{D}\rvert$	**Scalability Index Γ**						
		U	$\tau^{\mathfrak{d}}_{TTS}$	τ^{ψ}_{TTS}	**C**	**M**	**B**	**D**
$V = 20$	2	0.68	0.90	1.92	1.21	2.37	3.08	1.93
$\lambda = 0.033$	4	0.45	0.79	1.71	1.12	4.04	4.35	1.95
$V = 40$	2	0.62	0.60	1.27	1.30	2.11	3.93	2.00
$\lambda = 0.066$	4	0.60	0.45	0.98	1.92	3.52	7.88	3.49

First, the obtained results show that distribution contributes to improved scalability regarding domain paths setup (20% and 55% TTS decrease for two- and four-domain H-SDN compared to C-SDN). Coordination across SDNCs significantly impacts inter-domain path setup under small load (70-90% higher TTS for H-SDN), but This effect nearly disappears for larger topology and heavier loads (27 % TTS increase for two domains, and 2% TTS decrease for 4 domains), confirming superior scalability of H-SDN.

In terms of resources, Γ is favorable in most cases except bandwidth and disk I/O. CPU usage grows sub-linearly with the number of SDNCs (less than doubled in the extreme case), and memory consumption scales roughly linearly with SDNCs count. In contrast, bandwidth and disk I/O grow non-linearly with SDNC load and domain count. Up to 8 times higher bandwidth than in C-SDN case has been measured, which can notably increase H-SDN deployment costs in large-scale networks.

Overall, distributing DP control across multiple SDNCs increases total CP capacity, unlike C-SDN, where a single SDNC may collapse despite unused resources available for scaling (cf. Fig. 6e and Fig. 8 for $\lambda = 0.066$). H-SDN also lowers TTS for intra-domain flows thanks to the proximity of controllers and switches and reduced RTT, though more complex exchanges between SDNCs and HSDNC and queuing, impacts TTS jitter and variance. Moreover, scalability benefits depend on workloads distribution across SDNCs with respect to their service rates. Otherwise, a congestion in a single domain will impact not only domain flows but also inter-domain ones that enter/exit this domain (e.g., $V = 20, \lambda = 0.066, H = 50$; cf. Fig. 6).

The practical deployment of production-grade H-SDN still faces several unaddressed issues that affect CP scalability and pose SDN CP design challenges. The main challenges encountered in carrier-grade SDN networks are outlined below.

Topology heterogeneity – While H-SDN alleviates technological heterogeneity across network domains, achieving balanced workload distribution among SDNCs is rarely feasible due to diversified network topologies, ownership, user distribution, or applications placement. Consequently, the network planning requires careful analysis and capacity planning (e.g., AI-based methods) to anticipate traffic volumes and avoid domain-level bottlenecks that could degrade E2E performance for flows originating, terminating, or traversing the given domain.

Degree of CP distribution – CP distribution introduces E2E coordination overheads that may nullify scalability benefits as the number of domains grows [95], similar to synchronization errors in multi-processor systems [96]. Likewise, in C-SDN, scaling up an SDNC instance can increases the service rate (via concurrency), only until software-level context-switching overhead becomes dominant. As shown in Fig. 8, SDNC can saturate even when resources are available for the instance (e.g., SDNC overload while $C_{total} < 10\%$). Moreover, redundancy and optimal placement of hierarchical controllers is essential for H-SDN to enable timely synchronization, load balancing and failover at E2E level. Such strategies are still not well investigated in the literature and remain a challenge.

Deployment costs – H-SDN incurs higher deployment costs due to additional compute, network, and storage resources required for multiple controller instances and accommodating SDNC-HSDNC exchange. These costs are particularly critical in resource-constrained environments with expensive infrastructure up-scaling, such as Non-Terrestrial Networks (NTNs). Hierarchical control also increases CP traffic for topology management and monitoring. This issue can be alleviated by leveraging metrics, network abstractions and AI-driven agents for decision making [97].

Monitoring overhead – Ensuring E2E QoS, requires fine-grained monitoring of flows, which can impose substantial CP load and lower the capacity to serve user traffic. This issue is especially problematic due to burstiness that can periodically saturate the CP, increasing TTS. This issue will be further aggravated in volatile topologies like NTN, due to the changing number of switches and flows assigned to SDNC, additionally incurring discovery traffic overheads. Therefore, flow-oriented monitoring approaches should be adopted, such as In-band Telemetry [48], to avoid excessive interactions between SDNC and DP devices.

Fault management & failover – Failover in H-SDN requires rerouting of the flows traversing the failing link (switch) to a non-failing network path meeting specific QoS demands and can involve one or more domain SDNCs and HSDNC to recompute paths and install relevant flow entries (generating CP load peaks). If domain SDNC fails, the control path recovery needs to be performed usually involving reassigning the domain switches to neighboring SDNCs. The control path reestablishment in multi-controller environments is an NP hard problem, and is still not well examined [98].

E2E abstraction and path optimality – A central aspect of H-SDN is the use of network abstractions for E2E control and offloading upper-layer controllers [99]. The abstraction fidelity is critical to balance control delegation to domain SDNCs, while preserving sufficient global visibility for optimal and valid E2E paths enforcement.

State consistency and trust – Feployment of multiple SDNCs operating under different administrative domains requires both state synchronization and trusted access control. Distributed ledger technologies can facilitate address this issue by providing trustworthy and coordinated data access for SDNCs [100].

Network dimensioning – The actual SDN CP deployments require careful dimensioning of the number of controllers and allocated resources (cf. Section 4.2) for optimal usage, while maintaining desired flow serving capacity and speed (e.g., TTS). The model proposed in this paper can support such cases by: i) providing analytical method for estimation of controller utilization and forecasts of TTS; ii) enabling assessment of resource consumption growth in the function of controller utilization. These capabilities can thus facilitate network dimensioning both the domain and E2E levels.

6 Future work

This study introduced a foundation scalability model for various SDN CP architectures and presented its initial evaluation. Future work will extend the model assessment and incorporate ζ overhead modeling under highly dynamic conditions, such as in NTN environments. Further validation will invovle diverse SDNC implementations (e.g., OpenDaylight, ONOS, TerraFlow) in large-scale, heterogeneous topologies with varying traffic patterns, and failover scenarios. We also plan to compare model accuracy across decentralized cloud (telco cloud) and on-premises deployments, and to extend the scalability model toward the NTN case. In such environments, mobility of nodes and topology changes introduce additional temporal CP overheads (e.g., topology discovery, flow rerouting, and dynamic controller assignment), motivating data-driven or AI-assisted modeling of ζ. Finally, we aim to study the effect of abstraction fidelity on path optimality to support optimized design decisions for H-SDN architectures.

7 Summary and conclusions

Due to its high adaptability, Software-Defined Networking (SDN) can deliver substantial value to modern telecommunication systems, including data center networks, virtualized environments, and mobile networks. The SDN incorporation, however, requires careful evaluation and addressing the CP scalability challenges. In this paper, we have proposed a Flatto–Hahn–Wright (FHW)-based model for estimating key CP performance metrics related to CP scalability: Time-To-Serve (TTS) and resource utilization of CP components. In the analyses, we focused on SDN operating in reactive mode and the most popular SDN architecture variants: Centralized SDN (C-SDN) and Hierarchical multi-controller SDN (H-SDN). We performed foundational validation of the models through experiments using a virtualized SDN emulation testbed based on open-source SDN tools and have discussed the usefulness of the models in designing C-SDN and H-SDN networks. The model can be leveraged to build tools for network dimensioning and performance assessment to support predictive network maintenance. With the introduced scalability index, we show that H-SDN achieves better scalability, yielding over 50% lower TTS for intra-domain flows while keeping a similar TTS for inter-domain flows. Moreover, we discussed the resource scalability of the two variants. While H-SDN exhibits a very small increase in CPU consumption compared to C-SDN, it requires significantly more memory, bandwidth, and disk I/O speed to operate, which can potentially induce high additional costs. Finally, we outlined key challenges in deploying production-grade SDN systems related to CP scalability and discussed future research directions for improving scalability modeling in multi-controller architectures.

Declarations

Funding: ETHER project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101096526. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Conflicts of interest: The authors have no conflicts of interest to declare that are relevant to the content of this article.

Data availability: The data that support the findings of this study are available from the corresponding author upon reasonable request. Availability of these data may be subject to restrictions due to their size and storage constraints.

References

[1] Ammar, S., Pong Lau, C., Shihada, B.: An In-Depth Survey on Virtualization Technologies in 6G Integrated Terrestrial and Non-Terrestrial Networks. IEEE Open Journal of the Communications Society **5**, 3690–3734 (2024) https://doi.org/10.1109/OJCOMS.2024.3414622

[2] Gómez-delaHiz, J., Galán-Jiménez, J.: Improving the Traffic Engineering of SDN Networks by Using Local Multi-Agent Deep Reinforcement Learning. In: NOMS 2024-2024 IEEE Network Operations and Management Symposium, pp. 1–5. IEEE, Seoul, Korea, Republic of (2024). https://doi.org/10.1109/NOMS59830.2024.10575465

[3] Long, Q., Chen, Y., Zhang, H., Lei, X.: Software Defined 5G and 6G Networks: A Survey. Mobile Networks and Applications **27**(5), 1792–1812 (2022) https://doi.org/10.1007/s11036-019-01397-2

[4] Bouras, C., Kollia, A., Papazois, A.: SDN & NFV in 5G: Advancements and Challenges. In: 2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN), pp. 107–111. IEEE, Paris (2017). https://doi.org/10.1109/ICIN.2017.7899398

[5] Hantouti, H., Benamar, N., Taleb, T.: Service Function Chaining in 5G & Beyond Networks: Challenges and Open Research Issues. arXiv (2022). https://doi.org/10.48550/arXiv.2201.05455

[6] Han, L., Retana, A., Westphal, C., Li, R.: Large Scale LEO Satellite Networks for the Future Internet: Challenges and Solutions to Addressing and Routing. Computer Networks and Communications (2022) https://doi.org/10.37256/cnc.1120232105

[7] Bojović, P.D., Malbašić, T., Vujošević, D., Martić, G., Bojović, Ž.: Dynamic QoS Management for a Flexible 5G/6G Network Core: A Step toward a Higher Programmability. Sensors **22**(8), 2849 (2022) https://doi.org/10.3390/s22082849

[8] Mai, V.S., La, R.J., Zhang, T., Battou, A.: End-to-End Quality-of-Service Assurance with Autonomous Systems: 5G/6G Case Study. arXiv (2022). https://doi.org/10.48550/arXiv.2201.13300

[9] Kanwal, A., Nizamuddin, M., Iqbal, W., Aman, W., Abbas, Y., Mussiraliyeva, S.: Exploring Security Dynamics in SDN Controller Architectures: Threat Landscape and Implications. IEEE Access **12**, 56517–56553 (2024) https://doi.org/10.1109/ACCESS.2024.3390968

[10] Das, T., Sridharan, V., Gurusamy, M.: A Survey on Controller Placement in SDN. IEEE Communications Surveys & Tutorials **22**(1), 472–503 (2020) https:

//doi.org/10.1109/COMST.2019.2935453

[11] Blial, O., Ben Mamoun, M., Benaini, R.: An Overview on SDN Architectures with Multiple Controllers. Journal of Computer Networks and Communications **2016**(1), 9396525 (2016) https://doi.org/10.1155/2016/9396525

[12] Koponen, T., Casado, M., Gude, N., Stribling, J., Poutievski, L., Zhu, M., Ramanathan, R., Iwata, Y., Inoue, H., Hama, T., Shenker, S.: Onix: A Distributed Control Platform for Large-Scale Production Networks. In: Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation. OSDI'10, pp. 351–364. USENIX Association, USA (2010)

[13] Phemius, K., Bouet, M., Leguay, J.: DISCO: Distributed SDN Controllers in a Multi-Domain Environment. In: 2014 IEEE Network Operations and Management Symposium (NOMS), pp. 1–2 (2014). https://doi.org/10.1109/NOMS.2014.6838273

[14] Dixit, A., Hao, F., Mukherjee, S., Lakshman, T.V., Kompella, R.: Towards an Elastic Distributed SDN Controller. SIGCOMM Comput. Commun. Rev. **43**(4), 7–12 (2013) https://doi.org/10.1145/2534169.2491193

[15] ETSI: TeraFlowSDN. https://www.teraflow-h2020.eu/

[16] Fu, Y., Bi, J., Gao, K., Chen, Z., Wu, J., Hao, B.: Orion: A Hybrid Hierarchical Control Plane of Software-Defined Networking for Large-Scale Networks. In: 2014 IEEE 22nd International Conference on Network Protocols, pp. 569–576 (2014). https://doi.org/10.1109/ICNP.2014.91

[17] Hassas Yeganeh, S., Ganjali, Y.: Kandoo: A Framework for Efficient and Scalable Offloading of Control Applications. In: Proceedings of the First Workshop on Hot Topics in Software Defined Networks, pp. 19–24. ACM, Helsinki Finland (2012). https://doi.org/10.1145/2342441.2342446

[18] Ballotta, L., Jovanović, M.R., Schenato, L.: Can Decentralized Control Outperform Centralized? The Role of Communication Latency. IEEE Transactions on Control of Network Systems **10**(3), 1629–1640 (2023) https://doi.org/10.1109/TCNS.2023.3237483 arXiv:2109.00359 [math]

[19] Hu, J., Lin, C., Li, X., Huang, J.: Scalability of Control Planes for Software Defined Networks: Modeling and Evaluation. In: 2014 IEEE 22nd International Symposium of Quality of Service (IWQoS), pp. 147–152. IEEE, Hong Kong (2014). https://doi.org/10.1109/IWQoS.2014.6914314

[20] Ravuri, H.K., Vega, M.T., Van Der Hooft, J., Wauters, T., Da, B., De Turck, F.: On Routing Scalability in Flat SDN Architectures. In: 2020 11th International Conference on Network of the Future (NoF), pp. 23–27. IEEE, Bordeaux, France (2020). https://doi.org/10.1109/NoF50125.2020.9249170

[21] Karakus, M., Durresi, A.: A Survey: Control Plane Scalability Issues and Approaches in Software-Defined Networking (SDN). Computer Networks **112**, 279–293 (2017) https://doi.org/10.1016/j.comnet.2016.11.017

[22] Yeganeh, S.H., Tootoonchian, A., Ganjali, Y.: On Scalability of Software-Defined Networking. IEEE Communications Magazine **51**(2), 136–141 (2013) https://doi.org/10.1109/MCOM.2013.6461198

[23] Zuo, Q., Chen, M., Ding, K., Xu, B.: On generality of the data plane and scalability of the control plane in software-defined networking. China Communications **11**(2), 55–64 (2014) https://doi.org/10.1109/CC.2014.6821737

[24] Wang, J., Shou, G., Hu, Y., Guo, Z.: A Multi-Domain SDN Scalability Architecture Implementation Based on the Coordinate Controller. In: 2016 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), pp. 494–499. IEEE, Chengdu, China (2016). https://doi.org/10.1109/CyberC.2016.100

[25] Karakus, M., Durresi, A.: A Scalability Metric for Control Planes in Software Defined Networks (SDNs). In: 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA), pp. 282–289. IEEE, Crans-Montana, Switzerland (2016). https://doi.org/10.1109/AINA.2016.112

[26] Ntontin, K., Tomaszewski, L., Ruiz-de-Azua, J.A., Cárdenas, A., Centelles, R.P., Lin, C.-K., Mesodiakaki, A., Antonopoulos, A., Pappas, N., Fiore, M., Aguilar, S., Watts, S., Harris, P., Santiago, A.R., Lazarakis, F., Calisti, M., Chatzinotas, S.: ETHER: A 6G Architectural Framework for 3D Multi-Layered Networks. In: 2024 IEEE Wireless Communications and Networking Conference (WCNC), pp. 1–6. IEEE, Dubai, United Arab Emirates (2024). https://doi.org/10.1109/WCNC57260.2024.10570731

[27] Chen, C., Liao, Z., Ju, Y., He, C., Yu, K., Wan, S.: Hierarchical Domain-Based Multicontroller Deployment Strategy in SDN-Enabled Space–Air–Ground Integrated Network. IEEE Transactions on Aerospace and Electronic Systems **58**(6), 4864–4879 (2022) https://doi.org/10.1109/TAES.2022.3199191

[28] Liao, Z., Chen, C., Ju, Y., He, C., Jiang, J., Pei, Q.: Multi-Controller Deployment in SDN-Enabled 6G Space–Air–Ground Integrated Network. Remote Sensing **14**(5), 1076 (2022) https://doi.org/10.3390/rs14051076

[29] Kuklinski, S., Li, Y., Dinh, K.T.: Handover management in SDN-based mobile networks. In: 2014 IEEE Globecom Workshops (GC Wkshps), pp. 194–200. IEEE, Austin, TX, USA (2014). https://doi.org/10.1109/GLOCOMW.2014.7063430

[30] Zobary, F., Li, C.L.: A Mathematical Model for SDN Control Plane Scalability

Evaluation Based on Controller Utilization. Journal of Control Engineering and Applied Informatics **25** (2023) https://doi.org/10.61416/ceai.v25i4.8783

[31] Foundation, O.N.: SDN Architecture. Technical Reference TR-502, ONF (June 2014)

[32] Foundation, O.N.: OpenFlow Switch Specification; Version 1.5.1 (Protocol version 0x06). Technical Specification TS-025, Open Networking Foundation (March 2015)

[33] Kukkalli, H., Hajizadeh, M., Bauschert, T.: Practical Evaluation of Dynamic Service Function Chaining (SFC) for Softwarized Mobile Services in an SDN-based Cloud Network. In: NOMS 2024-2024 IEEE Network Operations and Management Symposium, pp. 1–6. IEEE, Seoul, Korea, Republic of (2024). https://doi.org/10.1109/NOMS59830.2024.10575344

[34] Abuarqoub, A.: A Review of the Control Plane Scalability Approaches in Software Defined Networking. Future Internet **12**(3) (2020) https://doi.org/10.3390/fi12030049

[35] Bannour, F., Souihi, S., Mellouk, A.: Distributed SDN Control: Survey, Taxonomy, and Challenges. IEEE Communications Surveys & Tutorials **20**(1), 333–354 (2018) https://doi.org/10.1109/COMST.2017.2782482

[36] Khorsandroo, S., Sánchez, A.G., Tosun, A.S., Arco, JM., Doriguzzi-Corin, R.: Hybrid SDN Evolution: A Comprehensive Survey of the State-of-the-Art. Computer Networks **192**, 107981 (2021) https://doi.org/10.1016/j.comnet.2021.107981

[37] Fernandez, M.P.: Comparing OpenFlow Controller Paradigms Scalability: Reactive and Proactive. In: 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA), pp. 1009–1016. IEEE, Barcelona (2013). https://doi.org/10.1109/AINA.2013.113

[38] Nguyen, X.-N., Saucez, D., Barakat, C., Turletti, T.: Rules Placement Problem in OpenFlow Networks: A Survey. IEEE Communications Surveys & Tutorials **18**(2) (2016)

[39] McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., Turner, J.: OpenFlow: Enabling innovation in campus networks. SIGCOMM Comput. Commun. Rev. **38**(2), 69–74 (2008) https://doi.org/10.1145/1355734.1355746

[40] Guan, X., Choi, B.-Y., Song, S.: Reliability and Scalability Issues in Software Defined Network Frameworks. In: 2013 Second GENI Research and Educational Experiment Workshop, pp. 102–103. IEEE, Salt Lake, UT, USA (2013). https://doi.org/10.1109/GREE.2013.28

[41] Iyer, A.S., Mann, V., Samineni, N.R.: SwitchReduce: Reducing switch state and controller involvement in OpenFlow networks. In: 2013 IFIP Networking Conference, pp. 1–9 (2013)

[42] Cai, Z., Cox, A.L., Ng, T.S.E.: Maestro: A System for Scalable OpenFlow Control. Computer Science Technical Reports (2010)

[43] SR, A., Mahadev, K., Prasad, S., Eswaran, S., Honnavalli, P.: OpenDaylight as Software Defined Networking Controller: Shortcomings and Possible Solutions. In: 2022 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), pp. 1–6 (2022). https://doi.org/10.1109/CONECCT55679.2022.9865784

[44] Berde, P., Gerola, M., Hart, J., Higuchi, Y., Kobayashi, M., Koide, T., Lantz, B., O'Connor, B., Radoslavov, P., Snow, W., Parulkar, G.: ONOS: Towards an open, distributed SDN OS. In: Proceedings of the Third Workshop on Hot Topics in Software Defined Networking. HotSDN '14, pp. 1–6. Association for Computing Machinery, New York, NY, USA (2014). https://doi.org/10.1145/2620728.2620744

[45] Kubo, R., Fujita, T., Agawa, Y., Suzuki, H.: Ryu SDN Framework-Open-Source SDN Platform Software. NTT Technical Review **12** (2014) https://doi.org/10.53829/ntr201408fa4

[46] Bosshart, P., Daly, D., Gibb, G., Izzard, M., McKeown, N., Rexford, J., Schlesinger, C., Talayco, D., Vahdat, A., Varghese, G., Walker, D.: P4: Programming Protocol-Independent Packet Processors. ACM SIGCOMM Computer Communication Review **44**(3), 87–95 (2014) https://doi.org/10.1145/2656877.2656890

[47] Paolucci, F., Civerchia, F., Sgambelluri, A., Giorgetti, A., Cugini, F., Castoldi, P.: P4 Edge Node Enabling Stateful Traffic Engineering and Cyber Security. Journal of Optical Communications and Networking **11**(1), 84–95 (2019)

[48] Bae, C., Lee, K., Kim, H., Yoon, S., Hong, J., Pack, S., Lee, D.: Quantized In-band Network Telemetry for Low Bandwidth Overhead Monitoring. In: 2024 20th International Conference on Network and Service Management (CNSM), pp. 1–5. IEEE, Prague, Czech Republic (2024). https://doi.org/10.23919/CNSM62983.2024.10814621

[49] Varyani, N., Zhang, Z.-L., Dai, D.: QROUTE: An Efficient Quality of Service (QoS) Routing Scheme for Software-Defined Overlay Networks. IEEE Access **8**, 104109–104126 (2020) https://doi.org/10.1109/ACCESS.2020.2995558

[50] Prabha, C., Goel, A., Singh, J.: A Survey on SDN Controller Evolution: A Brief Review, shorttitle = A Survey on SDN Controller Evolution. In: 2022 7th International Conference on Communication and Electronics Systems

(ICCES), pp. 569–575. IEEE, Coimbatore, India (2022). https://doi.org/10.1109/ICCES54183.2022.9835810

[51] Networks, N., University, S., International Computer Science Institute, U.B.: NOX Network Control Platform (2025)

[52] McCauley, J., Scott, C., Wundsam, A., Guedes, D., Networks, N.: POX Controller (2025)

[53] Wang, Q., Barrineau, G., Izard, R., Parraga, J., Wang, K.-C.K., Ramasubramanian, S.: Floodlight OpenFlow Controller (OSS). Project Floodlight (2025)

[54] Vizarreta, P., Trivedi, K., Mendiratta, V., Kellerer, W., Mas-Machuca, C.: DASON: Dependability Assessment Framework for Imperfect Distributed SDN Implementations. IEEE Transactions on Network and Service Management **17**(2), 652–667 (2020) https://doi.org/10.1109/TNSM.2020.2973925

[55] Zhang, H., Yan, J.: Performance of SDN Routing in Comparison with Legacy Routing Protocols. In: 2015 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, pp. 491–494. IEEE, Xi'an, China (2015). https://doi.org/10.1109/CyberC.2015.30

[56] Yahyaoui, H., Zhani, M.F.: On Providing Low-cost Flow Monitoring for SDN Networks. In: 2020 IEEE 9th International Conference on Cloud Networking (CloudNet), pp. 1–6. IEEE, Piscataway, NJ, USA (2020). https://doi.org/10.1109/CloudNet51028.2020.9335797

[57] Nougnanke, K.B., Bruyere, M., Labit, Y.: Low-Overhead Near-Real-Time Flow Statistics Collection in SDN. In: 2020 6th IEEE Conference on Network Softwarization (NetSoft), pp. 155–159. IEEE, Ghent, Belgium (2020). https://doi.org/10.1109/NetSoft48620.2020.9165421

[58] Malik, A., De Fréin, R.: Graph Modeling for OpenFlow Switch Monitoring. IEEE Access **11**, 84543–84553 (2023) https://doi.org/10.1109/ACCESS.2023.3303847

[59] Azzouni, A., Mai Trang, N.T., Boutaba, R., Pujolle, G.: Limitations of Openflow Topology Discovery Protocol. In: 2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), pp. 1–3. IEEE, Budva, Montenegro (2017). https://doi.org/10.1109/MedHocNet.2017.8001642

[60] Wazirali, R., Ahmad, R., Alhiyari, S.: SDN-OpenFlow Topology Discovery: An Overview of Performance Issues. Applied Sciences **11**(15), 6999 (2021) https://doi.org/10.3390/app11156999

[61] Kfoury, E.F., Crichigno, J., Bou-Harb, E.: An Exhaustive Survey on P4 Programmable Data Plane Switches: Taxonomy, Applications, Challenges, and Future Trends. IEEE Access **9**, 87094–87155 (2021) https://doi.org/10.1109/ACCESS.2021.3086704

[62] Sun, S., Wang, M., Zhou, L., Dong, J.: Load Balancing Algorithm for SDN Multi-Controller with Elite Genetic Algorithm. In: 2023 IEEE 6th International Conference on Electronics and Communication Engineering (ICECE), pp. 146–150. IEEE, Xi'an, China (2023). https://doi.org/10.1109/ICECE59822.2023.10462299

[63] Mokhtar, H., Di, X., Jiang, Z., Hamdan, M.: Dynamic Controller Assignment in LEO Satellite Based SDN Using Multi-Agent Deep Reinforcement Learning. In: 2024 IEEE Conference on Standards for Communications and Networking (CSCN), pp. 208–211. IEEE, Belgrade, Serbia (2024). https://doi.org/10.1109/CSCN63874.2024.10849764

[64] Saeed, K., Ullah, M.O.: Toward Reliable Controller Placements in Software-Defined Network Using Constrained Multi-Objective Optimization Technique. IEEE Access **10**, 129865–129883 (2022) https://doi.org/10.1109/ACCESS.2022.3228039

[65] Kondo, T., Guillen, L., Izumi, S., Abe, T., Suganuma, T.: Energy-Aware Controller Placement With Delay Guarantees for In-Band SDN. International Journal of Network Management **36**(1), 70036 (2026) https://doi.org/10.1002/nem.70036

[66] Maity, I., Ntontin, K., Chatzinotas, S.: CoMOSat: Controller Placement in SDN-Enabled Multi-Orbit Satellite Networks. In: 2025 IEEE International Conference on Communications Workshops (ICC Workshops), pp. 2026–2032. IEEE, Montreal, QC, Canada (2025). https://doi.org/10.1109/ICCWorkshops67674.2025.11162452

[67] Das, T., Gurusamy, M.: Controller Placement for Resilient Network State Synchronization in Multi-Controller SDN. IEEE Communications Letters **24**(6), 1299–1303 (2020) https://doi.org/10.1109/LCOMM.2020.2979072

[68] Huang, V., Chen, G., Zhang, P., Li, H., Hu, C., Pan, T., Fu, Q.: A Scalable Approach to SDN Control Plane Management: High Utilization Comes With Low Latency. IEEE Transactions on Network and Service Management **17**(2), 682–695 (2020) https://doi.org/10.1109/TNSM.2020.2973222

[69] Yahyaoui, H., Zhani, M.F., Bouachir, O., Aloqaily, M.: On minimizing flow monitoring costs in large-scale software-defined network networks. International Journal of Network Management **33**(2), 2220 (2023) https://doi.org/10.1002/nem.2220 https://onlinelibrary.wiley.com/doi/pdf/10.1002/nem.2220

[70] Hung, M.-H., Teng, C.-C., Chuang, C.-P., Hsu, C.-S., Gong, J.-W., Chen, M.-C.: A SDN Controller Monitoring Architecture for 5G Backhaul Networks. In: 2022 23rd Asia-Pacific Network Operations and Management Symposium (APNOMS), pp. 1–4. IEEE, Takamatsu, Japan (2022). https://doi.org/10.23919/APNOMS56106.2022.9919988

[71] Gharbaoui, M., Martini, B., Castoldi, P.: A WAN Infrastructure Manager Prototype for QoS-based Multi-Site SDN-Enabled NFV Orchestration. Journal of Network and Systems Management **33**(3), 54 (2025) https://doi.org/10.1007/s10922-025-09929-w

[72] Frate, M., Marczuk, M.K., Verdi, F.L.: OrchFlow: An Architecture for Orchestration of Multiple Controllers in OpenFlow Networks. Journal of Network and Systems Management **27**(3), 551–572 (2019) https://doi.org/10.1007/s10922-018-9476-x

[73] Shahriyar, M.M., Saha, G., Bhattacharjee, B., Reaz, R.: DEFT: Distributed, Elastic, and Fault-Tolerant State Management of Network Functions. In: 2023 19th International Conference on Network and Service Management (CNSM), pp. 1–7 (2023). https://doi.org/10.23919/CNSM59352.2023.10327813

[74] Lokesh, B., Rajagopalan, N.: Orchestrator for Synchronizing Network Events in SDNs. IEEE Transactions on Network and Service Management **18**(4), 4365–4375 (2021) https://doi.org/10.1109/TNSM.2021.3095060

[75] Serna, C.B., Mas-Machuca, C.: Preventing Control Plane Overload in SDN Networks with Programmable Data Planes. In: 2022 18th International Conference on Network and Service Management (CNSM), pp. 37–45 (2022). https://doi.org/10.23919/CNSM55787.2022.9964491

[76] Ren, Q., Guo, Z., Wu, J., Hu, T., Jie, L., Hu, Y., He, L.: SDN-ESRC: A Secure and Resilient Control Plane for Software-Defined Networks. IEEE Transactions on Network and Service Management **19**(3), 2366–2381 (2022) https://doi.org/10.1109/TNSM.2022.3163198

[77] Qiu, K., Huang, S., Xu, Q., Zhao, J., Wang, X., Secci, S.: ParaCon: A Parallel Control Plane for Scaling Up Path Computation in SDN. IEEE Transactions on Network and Service Management **14**(4), 978–990 (2017) https://doi.org/10.1109/TNSM.2017.2761777

[78] Yuan, D., Jiang, C., Zhao, Y., Yu, H., Zhu, Z.: A Hybrid Data Plane SDN Architecture Based on Cloud-Native for Segment Routing Over IPv6. IEEE Transactions on Network and Service Management **22**(5), 4619–4635 (2025) https://doi.org/10.1109/TNSM.2025.3588535

[79] Turlykozhayeva, D., Akhtanov, S., Ussipov, N., Akhmetali, A., Bolysbay, A., Shabdan, Y.: Routing Algorithm for Software Defined Network Based on

Boxcovering Algorithm. In: 2023 10th International Conference on Wireless Networks and Mobile Communications (WINCOM), pp. 1–5 (2023). https://doi.org/10.1109/WINCOM59760.2023.10322960

[80] Jia, Z., Sun, Y., Liu, Q., Dai, S., Liu, C.: cRetor: An SDN-Based Routing Scheme for Data Centers With Regular Topologies. IEEE Access **8**, 116866–116880 (2020) https://doi.org/10.1109/ACCESS.2020.3004609

[81] Zhu, L., Karim, M.M., Sharif, K., Li, F., Du, X., Guizani, M.: SDN Controllers: Benchmarking & Performance Evaluation. arXiv (2019). https://doi.org/10.48550/arXiv.1902.04491

[82] Tello, A.M.D., Abolhasan, M.: SDN Controllers Scalability and Performance Study. In: 2019 13th International Conference on Signal Processing and Communication Systems (ICSPCS), pp. 1–10. IEEE, Gold Coast, Australia (2019). https://doi.org/10.1109/ICSPCS47537.2019.9008462

[83] Sherwood, R., Yap, K., Sarrar, N., Erickson, D., Gibb, G., Wundsam, A., Huang, X., Tootoonchian, A.: Cbench: A Benchmarking Tool for Controllers. https://github.com/andi-bigswitch/oflops/tree/master/cbench

[84] Sattar, D., Matrawy, A.: An Empirical Model of Packet Processing Delay of the Open vSwitch. In: 2017 IEEE 25th International Conference on Network Protocols (ICNP), pp. 1–6 (2017). https://doi.org/10.1109/ICNP.2017.8117597

[85] Nougnanke, B., Labit, Y., Bruyere, M., Aïvodji, U., Ferlin, S.: ML-Based Performance Modeling in SDN-Enabled Data Center Networks. IEEE Transactions on Network and Service Management **20**(1), 815–829 (2023) https://doi.org/10.1109/TNSM.2022.3197789

[86] Wang, Y.: Research on the Queuing Theory Based on M/M/1 Queuing Model. Highlights in Science Engineering and Technology **61**, 80–87 (2025) https://doi.org/10.54097/hset.v61i.10276

[87] Wright, P.E.: Two Parallel Processors with Coupled Inputs. Advances in Applied Probability **24**(4), 986–1007 (1992) https://doi.org/10.2307/1427722

[88] Little, J.D.C.: A Proof for the Queuing Formula: L = λW. Operations Research **9**(3), 383–387 (1961) https://doi.org/10.1287/opre.9.3.383

[89] Docker: Docker: Accelerated Container Application Development. https://www.docker.com (2022)

[90] Lantz, B., Heller, B., McKeown, N.: A Network in a Laptop: Rapid Prototyping for Software-Defined Networks. In: Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks. Hotnets-IX, pp. 1–6. Association for Computing Machinery, New York, NY, USA (2010). https://doi.org/10.1145/

1868447.1868466

[91] Linux Foundation: Open vSwitch. https://www.openvswitch.org/

[92] VMWare Tanzu: RabbitMQ: One Broker to Queue Them All. https://www.rabbitmq.com

[93] Hagberg, A.A., Schult, D.A., Swart, P.J.: Exploring network structure, dynamics, and function using networkx. In: Proceedings of the 7th Python in Science Conference (SciPy2008), pp. 11–15 (2008)

[94] Dugan, J., Elliott, S., Mah, B.A., Poskanzer, J., Prabhu, K.: iPerf3. https://iperf.fr/iperf-download.php

[95] Abuarqoub, A.: A Review of the Control Plane Scalability Approaches in Software Defined Networking. Future Internet **12**(3), 49 (2020) https://doi.org/10.3390/fi12030049

[96] Perlin, N., Zysman, J.P., Kirtman, B.P.: Practical Scalability Assesment for Parallel Scientific Numerical Applications. arXiv (2016). https://doi.org/10.48550/arXiv.1611.01598

[97] Kołakowski, R., Kukliński, S., Tomaszewski, L.: Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Networks. In: 2024 IFIP Networking Conference (IFIP Networking), pp. 607–612 (2024). https://doi.org/10.23919/IFIPNetworking62109.2024.10619899

[98] Huang, F., Shi, J., Xv, Q.: Control path recovery after controller failure in SDN. In: 2023 8th International Conference on Intelligent Computing and Signal Processing (ICSP), pp. 1900–1904. IEEE, Xi'an, China (2023). https://doi.org/10.1109/ICSP58490.2023.10248805

[99] Ðerić, N., Varasteh, A., Basta, A., Blenk, A., Kellerer, W.: Sdn hypervisors: How much does topology abstraction matter? In: 2018 14th International Conference on Network and Service Management (CNSM), pp. 328–332 (2018)

[100] Díaz Rivera, J.J., Vilalta, R., Muñoz, R., Alemany, P., Renom, L.G.: Distributed Trust for Collaborative Network Management: Leveraging DLT in Multi-SDN Controller Environments. In: 2024 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), pp. 1–6 (2024). https://doi.org/10.1109/NFV-SDN61811.2024.10807463

Hierarchical Deep Reinforcement Learning-based Load Balancing Algorithm for Multi-domain Software-Defined Networks

Robert Kołakowski*◇, Sławomir Kukliński◇, Lechosław Tomaszewski*
*Orange Polska, ◇Warsaw University of Technology
Warsaw, Poland

***Abstract*—Software Defined Networking (SDN) is a well-established networking paradigm that enables granular network control and optimisation via Traffic Engineering (TE). A promising approach to SDN TE is to use centralised Deep Reinforcement Learning (DRL) enabling automated operation and optimisation both short and long-term. Despite excellent performance, the centralised DRL suffers from scalability and convergence issues, limiting its applicability. On the other hand, DRL exploitation in a multi-domain SDN environment is not well explored yet despite several benefits coming from operations distribution, such as better scalability or reduced impact of latency on Data Plane metrics collection. This paper presents the DRL-based routing approach targeting load balancing in a hierarchical multi-controller SDN. The concept yields network capacity gains over conventional routing methods. Apart from the improved scalability, the approach facilitates application in hybrid network deployments with limited interaction and visibility of domains' internals due to used abstractions of topology, metrics and path operations.**

***Index Terms*—6G, SDN, AI, DRL, User Plane, Data Plane, Control Plane, traffic engineering, load balancing, multi-agent system, DDPG, multi-domain routing**

I. Introduction

6G networks are commonly assumed to embed self-optimisation mechanisms in the future [1]. Software-Defined Network (SDN) is a well-established networking paradigm foreseen to play an important role in this context. The separation of Control Plane (CP) and Data Plane (DP) facilitates network traffic control and optimisation by Traffic Engineering (TE) applications. The classical SDN concept, due to centralisation, raises scalability issues in large networks. To solve this problem, distributed SDN architectures are often adopted [2], which complicate End-to-End (E2E) TE. The conventional TE methods, e.g., Open Shortest Path First (OSPF) [3], are not well suited for long-term optimisation, underperform in complex environments with rapid traffic fluctuations and lack the visibility of E2E network dynamics. A promising TE approach is to use Deep Reinforcement Learning (DRL), which dynamically adapts to variable conditions and is able to exploit complex environment properties.

Today, very few concepts combine DRL and distributed SDN to provide local and E2E optimisation simultaneously. Usually, a centralised model is adopted, which is unsuitable for carrier-grade networks due to poor scalability [4]. This paper proposes a novel multi-agent Deep Deterministic Policy Gradient (DDPG)-based routing algorithm, called Hierarchical Deep Reinforcement Learning Load Balancer (HDRL-LB), improving load distribution and total network capacity at both domain and global levels in a hierarchical multi-domain SDN. HDRL-LB introduces abstractions of topology, metrics and CP operations to support hybrid environments. The evaluation showed over 10% improvement of throughput compared to baseline state-of-the-art methods and fast policy convergence.

II. Related work

Future heterogeneous and distributed User Plane (UP) will require granular traffic control to provision E2E Quality of Service (QoS) [1]. TE will need to consider the current network state and imperceptible properties manifesting, e.g., in different traffic peaks seasonality, traffic types share, mobility patterns, etc. The DRL is promising in this context due to its ability to adapt to the environment traits by using the action-reward mechanism. The academia proposed several Reinforcement Learning (RL)/DRL frameworks for centralised SDN featuring automatic routing [5], [6] or load-balancing [7] to improve throughput and delay. The solutions, however, have not been tested in multi-domain SDN.

A routing concept for hierarchical multi-controller SDN has been proposed in [8]. The SDN Controllers (SDNCs) cooperate to find weighted shortest paths at the domain and global levels to avoid congestion. The collaborative multi-domain routing framework exploiting Integer Linear Programming (ILP) has been proposed in [9]. It ensures delay and bandwidth and maximises network usage. The DisTE approach provides max-min fair bandwidth allocation for flows and maximises

ETHER project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101096526. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.
ISBN 978-3-903176-63-8 © 2024 IFIP

978-3-903176-63-8/24/$31.00 ©2024 IEEE

resource usage in a multi-domain SDN [10]. The domains' synchronisation mechanism mitigates the selfish SDNCs' behaviour to obtain a consistent policy. The multi-agent cross-domain routing framework has been proposed in [11], which uses prediction to improve measurement reliability and DRL agents' performance.

Whereas TE in SDN is a well-known problem, the E2E optimisation of multi-domain SDN is not well addressed yet. Also, the scalability of SDNC operations is usually neglected. Hereby, we propose a scalable DRL-based TE algorithm that can be efficiently used in hierarchical multi-domain SDN setups.

III. Hierarchical Deep Reinforcement Learning Load Balancer (HDRL-LB)

A. Concept description

The key issues regarding wide-scale SDN implementation are SDN scalability and E2E TE. Distribution of control, while offloading each SDNCs and increasing scalability, increases the complexity of E2E TE. First, the optimisation is performed internally within each domain without a global view of its impact on the whole network. This can lead to a load imbalance on the inter-domain links and potentially to congestion and QoS degradation in the neighbouring domains. The uncoordinated approach to multi-domain TE can also lead to traffic imbalance in the network domains and inefficient resource usage. Second, reaching optimal local states does not imply achieving the global optimum. Therefore, to achieve E2E network optimisation in multi-domain SDN, there is a need to combine local and global TE to i) continuously search for a global optimum; ii) enable exploration of local search spaces; and iii) provide global optimisation without disruption of local operations.

The HDRL-LB algorithm addresses the above-mentioned issues. We adopt a multi-domain hierarchical SDN architecture (cf. Fig. 1) composed of i) Global SDN Controller (GSDNC) – a global network control entity that handles inter-domain operations (routing, metrics collection, E2E path enforcement by delegating path creation to respective SDNCs) using the network graph abstractions (cf. Section III-B); ii) domain SDNCs responsible for intra-domain operations; iii) agents responsible for the global- and domain-level optimisations, i.e., Global Load Balancing Agent (GLBA) and Domain Load Balancing Agents (DLBAs), respectively. The latter ones are responsible for the periodic calculation of routing graphs(cf. Sec. III-C), which are used by SDNC/GSDNC for routing and path enforcement (using SDN CP). Also, to stabilise the network and enable domain-level routing adaptation, the global routing graph is updated much less frequently than domain ones.

The components jointly provide the E2E routing and E2E TE targeting improvement of network load distribution and throughput (cf. Sec. III-C). To improve scalability, GSDNC sees only the abstracted view of the network composed of Border Nodes (BNs) – nodes connected to data sources/sinks or terminating inter-domain links, hosts, inter-domain links, and abstracted links – the connections between BNs pairs belonging to the same domain (cf. Fig. 1). Also, to reduce the number of CP operations and offload SDNCs, the domain and overlay routing graphs do not affect the already routed flows, only the new ones. This approach, combined with the ability to distribute the E2E routing across multiple SDNCs, contributes to the SDN scalability.

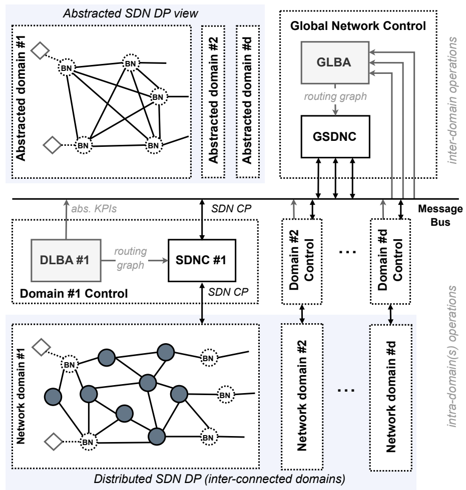


Fig. 1. HDRL-LB entities, interactions and DP views, i.e., network switches (circles) and hosts (rectangles), on each hierarchy level

When a new flow arrives in the network, SDNC identifies the routing case. If the target lies within domain boundaries, SDNC performs intra-domain routing. Otherwise, the flow metadata are forwarded to GSDNC, which performs the inter-domain routing. The output E2E path, in this case, is composed of BNs between the flow's source and target. Next, the path is split based on the BNs domain membership and enforced by respective SDNCs (performing intra-domain routing for the node pairs). In both cases, SDNCs and GSDNC compute the shortest paths using the Dijkstra algorithm on routing graphs with edge weights computed by agents.

To enable dynamic and autonomous UP optimisation, HDRL-LB uses modified DDPG [12], which uses Double Deep Q-Network (DDQN)-based [13] critic to mitigate initial over-optimism [14] and improve policy convergence. Its sample efficiency also contributes to SDN scalability by enabling less frequent DP sampling. In HDRL-LB, DLBAs exchanges with GLBA abstracted metrics. It supports the operation in hybrid multi-provider SDN environments, implementing different TE mechanisms within the domains. Hereby, we consider full operator's control over the underlying domains to assess the full HDRL-LB benefits. Finally, to leverage quasi-periodic demand peaks in the UP (i.e., busy hours) [15], we add time to the environment's state vector.

B. Network model

We consider the multi-domain network supervised by the centralised entity with a limited view of the domains' internals (cf. Section III-A). The topology is represented by an undirected graph $G(V, E)$, where V is a set of vertices and E is a set of edges. Hosts H (connected to vertices v of V) constitute traffic sources and sinks. We consider two network views: a global (abstracted) and domain one, denoted as $G^g(V^g, E^g)$ and $G^d(V^d, E^d)$ respectively ($d \in D$, where D is a set of domains). Domain switches combine a set $V^d = [v_1^d, v_2^d, ..., v_n^d]$ and edges a set $E^d = [e_{12}^d, e_{13}^d, ..., e_{ij}^d]$, where d denotes the node's domain and i, j the vertices v_i, v_j ($V^g = [v_1^g, v_2^g..., v_n^g]$ and $E^g = [e_{12}^g, e_{13}^g, ..., e_{ij}^g]$ for the global graph). Domain hosts constitute a set $H^d = [h_1^d, ..., h_m^d]$, where m is their number (all hosts in H^g case). The link connecting h_m^d with a switch has capacity c_m^d and bandwidth $\mathscr{b}_m^d$. All hosts are visible from the global view, i.e., $H^d \subset H^g$. In terms of vertices, the global network view is limited to the domain gateways (switches connected to hosts or inter-domain links), i.e., $V^g \subset V$, where $v_i^d \in V^g \ iff \ \exists_{v_j^d \in N(v_i)} v_j^d \in H^d \ \vee \ d_{v_i} \neq d_{v_j}$. The components at the global level see the abstracted edges (cf. Fig. 1). Each link e_{ij} has capacity c_{ij}, and handles aggregate traffic b_{ij} resulting in utilisation u_{ij} (eq. 1).

$$u_{ij} = \frac{b_{ij}}{c_{ij}} \tag{1}$$

For each domain, a set $U^d = [u_{12}^d, u_{13}^d, ...u_{ij}^d]$ describing the utilisation of the links E^d is defined ($U^g = [u_{12}^g, u_{13}^g, ...u_{ij}^g]$ for the global graph links E^g). The capacity of abstracted links observed by GLBA is calculated using f_{max} function (e.g., by using the Ford–Fulkerson algorithm) as shown in eq. 2.

$$c_{ij}^g = \begin{cases} c_{ij} & iff \ d_{v_i} \neq d_{v_j} \\ f_{max}(v_i, v_j) & iff \ d_{v_i} = d_{v_j} \end{cases} \tag{2}$$

The utilisation u_{ij}^g of the abstracted edge e_{ij}^g, is calculated by subtracting the current capacity under traffic (c^{gb}) from the nominal one (c^g) and normalisation, i.e., $u_{ij}^g = (c_{ij}^g - c_{ij}^{gb})/c_{ij}^g$. Each traffic flow is described by a tuple $f = (b, t, h_{src}, h_{dst})$, containing consumed bandwidth b (variable in time), flow duration t, source h_{src} and destination nodes h_{dst}.

C. Algorithm principles

HDRL-LB aims to solve the dynamic flow allocation problem to improve load distribution across the network links and domains and increase aggregate global and domain-level throughput. The optimised metrics are load balancing factor (eq. 3) and total host throughput (eq. 4).

$$\varphi = \sum_{i=1}^{n} \sum_{\substack{j=1 \\ i \neq j}}^{n} |u_{ij} - avg(U)| \tag{3}$$

$$\Gamma = \sum_{i=1}^{m} \mathscr{b}_i \tag{4}$$

The HDRL-LB target is the combination of conflicting goals for domain and global level entities (eq. 5 and 6).

$$\begin{aligned} min(\varphi^d) \\ max(\Gamma^d) \end{aligned} \tag{5}$$

$$\begin{aligned} min(\varphi^g) \\ max(\Gamma^g) \end{aligned} \tag{6}$$

We model the multi-domain hierarchical SDN network as the combination of standard RL settings (i.e., a set of stochastic domain environments constituting a stochastic global environment) and model each one as Markov Decision Process (MDP) represented by tuples $\mathcal{M} = (\mathcal{S}; \mathcal{A}; \mathcal{T}; \mathcal{R})$, where $\mathcal{S}$ – set of states s_t, $\mathcal{A}$ – set of actions a, $\mathcal{T}$ – transition probability from state s_t to s_{t+1} at time t after taking action a_t, $\mathcal{R}$ – reward function specifying reward r_t for s_t to s_{t+1} transition. We model the environment state s as the vector composed of utilisation u_{ij}^d of domain links (utilisation u_{ij}^g of abstracted links in GLBA case) and the normalised time $t_{norm} = (t_{cur} \mod t_{dur})/t_{dur}$, where t_{cur} is the current time and t_{dur} the duration of a day. Based on the state information, the DRL agents' output actions a_t – the routing graphs used by SDNCs/GSDNC. The agents' policies are evaluated using the reward functions shown in eq. 7 and eq. 8 (DLBAs and GLBA, respectively).

$$r^d = \Gamma^d - (\varphi^d + \varphi^{da}) \tag{7}$$

$$r^g = \Gamma^g - \varphi^g \tag{8}$$

Where φ^{da} denotes the abstracted domain load balancing factor (i.e., φ calculated using the single domain graph abstraction as perceived by GLBA) and aggregate throughput Γ^g is calculated as shown in eq. 9.

$$\Gamma^g = \sum_{d=1}^{D} \Gamma^d \tag{9}$$

GLBA is focused on the load distribution across the abstracted and inter-domain links and total throughput as both contribute to the network operator's profits (capacity and energy savings due to even load per node). The DLBA reward considers the load balancing factors of domain φ^d and the abstracted domain φ^{da}. The latter makes DLBA pursue not only the "selfish" domain-level goals but also the global ones by considering the load imbalance on domains' abstracted links. As DLBAs try to reach domain goals, GLBA stabilises the traffic distribution across the whole network, neglecting the impact of rapid intra-domain routing changes. The interactions between the agents are shown in Fig. 2.

The network state is acquired by dedicated entities. For DLBA, it is done by the Metrics Calculator (MC), which assembles state information s_t^d provided by SDNC

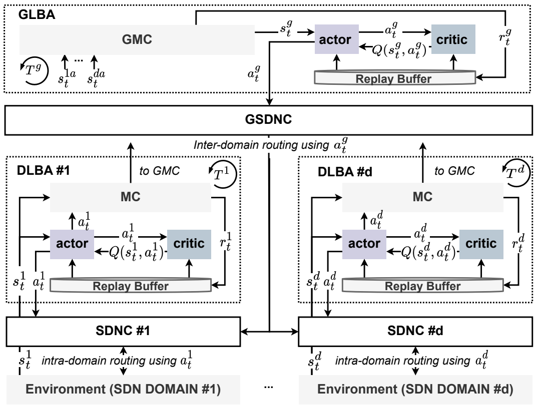


Fig. 2. Data exchange between the domain and global level entities

(i.a., $b_{ij}, c_{ij}, u_{ij}, \delta_m$) and calculates Γ^d and φ^d. The DLBA's actor consumes s_t^d to provide domain routing graph a_t^d. MC also derives the abstracted domain graph to obtain the abstracted domain load balancing factor φ^{da} for DLBA reward calculation. The abstracted domain graph is extended by the visible inter-domain links (with associated parameters) to obtain abstracted domain state s_t^{da} provided to Global Metrics Calculator (GMC). GMC joins the s_t^{da} received from each domain (eliminating link duplication) to get a global network state s_t^g, which is used to obtain rewards and global routing graph a_t^g, as in DLBA case. The routing graphs are sent to relevant SDNCs/GSDNC to be used for Dijkstra-based routing until the next update by DLBA/GLBA. Due to SDN specifics, to reduce SDNC load, the policy is updated every time interval T^x (T^d for DLBA, T^g for GLBA, $T^g >> T^d$), equal to the state sampling frequency. Each domain can, therefore, operate at a different time scale, as the network flows are routed using the current domains' routing graphs.

IV. Concept Evaluation and Results

To evaluate the HDRL-LB benefits, we conducted a set of tests verifying: i) performance gains under different network loads and traffic types; ii) the impact of the GLBA's overlay routing graph on the E2E performance. The tests were conducted under the Geant2019 [16] topology (40 nodes, 61 links), instantiated using the TopologyZoo dataset [17]. We divided topology into three domains (Fluid Communities algorithm), each managed and optimised by an SDNC/DLBA pair (cf. Fig. 3) and added a total of 15 hosts. Each link capacity was set to 20 Mbps. The influx of flow requests was modelled with the Poisson process with $\lambda = 3$ (20 flows per minute). Six test scenarios were conducted, corresponding to 50%, 75%, and 100% network loads for Transport Control Protocol (TCP) and User Datagram Protocol (UDP) traffic (cf. Table I), which was generated using iPerf3 [18] library. As the basis for comparison, we used Weighted Dijkstra (WD), which selects the best path based on the current link utilisation.

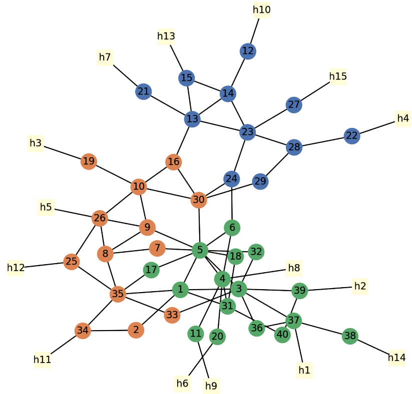


Fig. 3. GÉANT topology used for HDRL-LB evaluation [16] split into 3 domains, each handled by individual SDNC

TABLE I
Test Scenarios

Scenario	Flows Throughput [Mbps]	Avg. Flows Throughput [Mbps]	Type	Network Load [%]
SC1	$\mathcal{U}(5.2, 8.6)$	6.9	TCP	100
SC2	$\mathcal{U}(3.9, 6.5)$	5.2	TCP	75
SC3	$\mathcal{U}(2.6, 4.3)$	3.5	TCP	50
SC4	$\mathcal{U}(5.2, 8.6)$	6.9	UDP	100
SC5	$\mathcal{U}(3.9, 6.5)$	5.2	UDP	75
SC6	$\mathcal{U}(2.6, 4.3)$	3.5	UDP	50

Experiments were conducted in the hierarchical multi-domain SDN emulation using Mininet [19] with OpenFlow-enabled Open vSwitches [20], Ryu [21] SDNC and Python-based DLBA, GLBA, and GSDNC. The agents used Keras with TensorFlow back-end [22].

A. Performance

We evaluated the performance gains using the following metrics: capacity improvements (total volume conveyed by the network), network availability (number of served/missed flows) and user-perceived throughput (obtained using time- and episode-correlated client/server iPerf3 logs). Fig. 4 presents the improvements regarding the throughput Γ experienced by individual hosts.

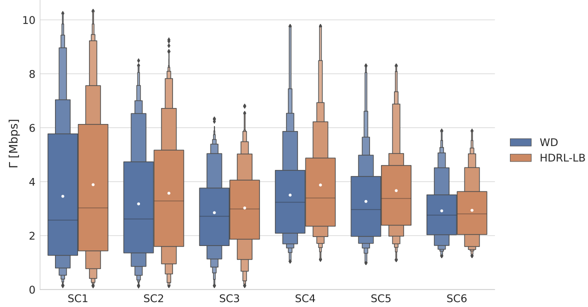


Fig. 4. Average Γ experienced by hosts (white dots – mean value)

In every case using HDRL-LB has led to improved average Γ. The first and third quartiles of Γ are higher for HDRL-LB than for the reference WD method. The general Γ improvement implies better throughput of individual network switches, linked with reduced buffering in the switches' ingress and egress queues. The latter indicates better load distribution across the network components improving the total carrying capacity. For the highest load scenarios, some values exceed the nominal maximum throughput specified in the test configurations (cf. Table I). This is due to the iPerf3 library configuration, which takes the average throughput as the input without the possibility of setting the upper bound of generated traffic for bursts in each interval. The results show similar trends for TCP and UDP traffic.

The biggest gains of HDRL-LB are achieved in the highest load scenarios (cf. Table II). Using HDRL-LB increases the aggregate data volume sent in the network (over 8.5% gain on average) and average throughput from 11.5% to almost 12.5% in the case of WD and TCP traffic (respectively, over 8% and 10.5% for UDP flows). Also, an almost 50% decrease of missed TCP flows can be observed. In UDP case, consecutive policy improvement led to a 30% decrease of flow drops after episode 50. In both cases, the improvement is achieved due to better load distribution and lowered congestion.

B. Convergence

In Fig. 5 and Fig. 6, the rewards obtained by the agents for all TCP and UDP traffic scenarios are presented.

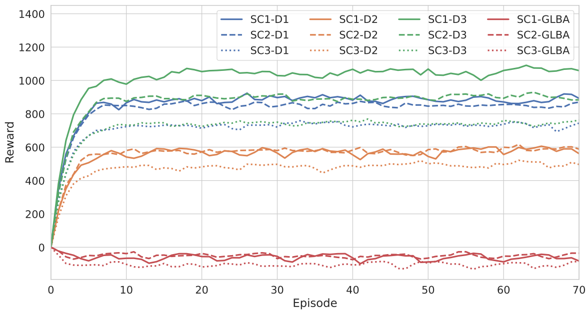


Fig. 5. Total rewards accumulated by the HDRL-LB agents across the training episodes under different TCP traffic loads

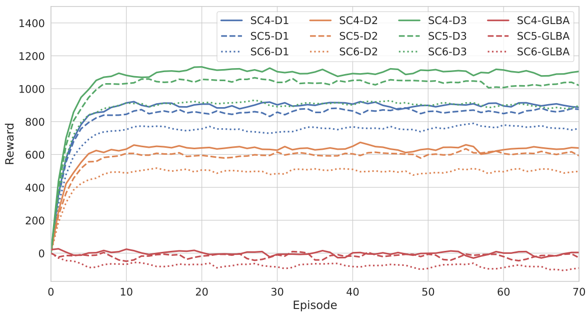


Fig. 6. Total rewards accumulated by the HDRL-LB agents across the training episodes under different UDP traffic loads

In every case, the convergence of the domain-level agents is relatively fast as the decent policy is reached after episode 20 with only minor improvements later on. The disproportions across the agents' rewards are linked with the domain topology, size, and the number of hosts. GLBA features the worst convergence due to the effects of conflicting domain-level policies. Nonetheless, a slight improvement during the training process can be seen. The poor stability of GLBA's learning is caused by the small number of domains corresponding to a limited number of states observed by the agent (and limited possibilities regarding load balancing across domains). Also, HDRL-LB uses the DDPG without convergence-oriented extensions, e.g., prioritised replay [7].

C. Impact of overlay routing graph

We evaluated the impact of using the overlay routing graph by running the highest load scenarios (SC1, SC4) without the operating GLBA, i.e., with the inter-domain routing performed using WD and intra-domain routing using modified DDPG with the same setup as in HDRL-LB case). The rewards accumulated by the agents in SC1 and SC4 are presented in Fig. 7 and Fig. 8, respectively.

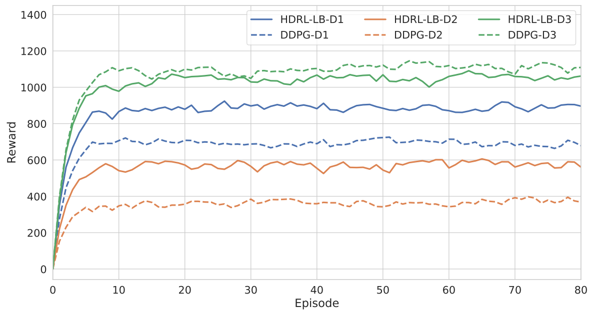


Fig. 7. Total rewards obtained for scenario SC1 using HDRL-LB algorithm vs. uncoordinated DDPG agents (without GLBA)

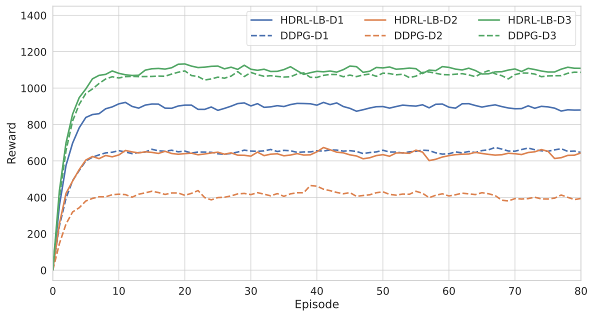


Fig. 8. Total rewards obtained for scenario SC4 using HDRL-LB algorithm vs. uncoordinated DDPG agents (without GLBA)

Similar policy convergence can be seen for supervised and non-supervised cases. While the global routing graph enforces some general rules on domain-level routing (i.e., selects the BNs), there is no apparent impact on the convergence of DLBAs policies. However, the following benefits of using the overlay routing graph can be observed (cf. Fig. 7 and Fig. 8):

- more even reward distribution across domains, which reflects efficient domain load balancing;
- higher aggregate reward for domain agents (SC1: around 50% gain for D1, D2 and around 10% for D3; SC4: 30% for D1, 50% D2, up to 10% for D3);
- higher total reward obtained by the agents.

TABLE II
CUMULATIVE PERFORMANCE COMPARISON OF HDRL-LB VS. WD ACROSS TRAINING EPISODES FOR HIGHEST LOAD SCENARIOS

Episode	Aggregate Data [GB]			Average Γ [Mbps]			Missed Flows [%]			SC
	HDRL-LB	WD	Gain [%]	HDRL-LB	WD	Gain [%]	HDRL-LB	WD	Gain [%]	
10	26.5	24.4	**8.61**	3.87	3.47	**11.6**	0.31	0.62	**50.0**	SC1 (TCP)
20	51.7	56.1	**8.51**	3.88	3.46	**11.9**	0.28	0.56	**50.0**	
30	78.7	85.7	**8.89**	3.89	3.46	**12.4**	0.25	0.55	**54.5**	
40	106.1	115.2	**8.58**	3.89	3.46	**12.4**	0.24	0.48	**50.0**	
50	133.3	144.8	**8.63**	3.89	3.46	**12.4**	0.24	0.45	**46.7**	
10	24.4	26.4	**8.2**	3.51	3.87	**10.3**	0.36	0.34	**-5.6**	SC4 (UDP)
20	51.5	55.7	**8.2**	3.51	3.88	**10.5**	0.33	0.32	**-3.0**	
30	78.7	85.1	**8.1**	3.51	3.87	**10.5**	0.30	0.33	**10.0**	
40	105.9	114.5	**8.1**	3.51	3.9	**10.6**	0.26	0.32	**23.1**	
50	133.1	143.8	**8.0**	3.50	3.9	**10.7**	0.24	0.32	**33.3**	

The above gains come solely from supervising GLBA and using network abstractions. The latter contributes to increased privacy and reusability as HDRL-LB can be applied in heterogeneous environments (comprising SDN and non-SDN domains) with limited access to the domains' internals and operations.

V. SUMMARY AND CONCLUSIONS

This paper presents a novel multi-agent DDPG-based routing algorithm for dynamic load-balancing in hierarchical multi-domain SDN (HDRL-LB). The domain-level optimisation is done at a fast scale to support rapid traffic fluctuation, while the global one stabilises the E2E network operation. The concept uses network abstractions to improve the scalability of SDN and TE and support multi-provider environments. The CP and TE distribution improves metrics accuracy (due to SDNC and switches collocation), convergence speed (reduced solution search spaces) and applicability in large networks. HDRL-LB also considers normalised time to improve the agents' behaviour during peak network traffic. The tests show that HDRL-LB yields significant throughput and carrying capacity gains compared to WD while reducing the number of missed flows. Future plans involve increasing HDRL-LB performance (GLBA policy convergence improvement), extensions towards fairness provisioning and tests in other topologies, while using various metric abstractions and reward functions.

REFERENCES

[1] T. Magedanz and M.-I. Corici, "Getting ready for 6G research – understanding technological drivers towards 6G and emerging 6G management requirements," Fraunhofer Fokus, Tutorial at IEEE/IFIP NOMS, April 25th, 2022, 2022, https://owncloud.fokus.fraunhofer.de/index.php/s/4oIylfMlANQTZCB/download.

[2] A. Abuarqoub, "A review of the control plane scalability approaches in software defined networking," *Future Internet*, vol. 12, no. 3, 2020, doi: 10.3390/fi12030049.

[3] IETF NWG, "OSPF Version 2," IETF, Tech. Rep., 1998. [Online]. Available: https://www.ietf.org/rfc/rfc2328.txt

[4] M. Karakus and A. Durresi, "A survey: Control plane scalability issues and approaches in software-defined networking (SDN)," *Computer Networks*, vol. 112, pp. 279–293, 2017, doi: 10.1016/j.comnet.2016.11.017.

[5] Y. Hu, Z. Li, J. Lan, J. Wu, and L. Yao, "EARS: Intelligence-driven experiential network architecture for automatic routing in software-defined networking," *China Communications*, vol. 17, no. 2, pp. 149–162, 2020, doi: 10.23919/JCC.2020.02.013.

[6] D. M. Casas-Velasco, O. M. C. Rendon, and N. L. S. da Fonseca, "Intelligent routing based on reinforcement learning for software-defined networking," *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 870–881, 2021, doi: 10.1109/TNSM.2020.3036911.

[7] J. Chen, Y. Wang, J. Ou, C. Fan, X. Lu, C. Liao, X. Huang, and H. Zhang, "ALBRL: Automatic load-balancing architecture based on reinforcement learning in software-defined networking," *Wireless Communications and Mobile Computing*, vol. 2022, pp. 1–17, May 2022, doi: 10.1155/2022/3866143.

[8] J.-J. Huang, Y.-Y. Chen, C. Chen, and Y. H. Chu, "Weighted routing in hierarchical multi-domain SDN controllers," in *2015 17th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, 2015, pp. 356–359, doi: 10.1109/APNOMS.2015.7275362.

[9] T. Moufakir, M. F. Zhani, A. Gherbi, and O. Bouachir, "Collaborative multi-domain routing in SDN environments," *Journal of Network and Systems Management*, vol. 30, Jan 2022, doi: 10.1007/s10922-021-09638-0.

[10] Y. Liu, L. Zhao, J. Hua, W. Qu, S. Zhang, and S. Zhong, "Distributed traffic engineering for multi-domain SDN without trust," *IEEE Transactions on Cloud Computing*, vol. 10, no. 4, pp. 2481–2496, 2022, doi: 10.1109/TCC.2021.3067456.

[11] M. Ye, L. Huang, X. Deng, Y. Wang, Q. Jiang, H. Qiu, and P. Wen, "A new intelligent cross-domain routing method in SDN based on a proposed multiagent reinforcement learning algorithm," 2023, doi: 10.48550/arXiv.2303.07572.

[12] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2019, doi: 10.48550/arXiv.1509.02971.

[13] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," 2015, doi: 10.48550/arXiv.1509.06461.

[14] H. van Hasselt, "Double Q-learning," in *Advances in Neural Information Processing Systems*, vol. 23. Curran Associates, Inc., 2010, pp. 2613–2621, doi: 10.5555/2997046.2997187.

[15] R. Kołakowski, S. Kukliński, and L. Tomaszewski, "Time-of-day-aware slice admission control," in *2023 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, 2023, pp. 199–204, doi: 10.1109/MeditCom58224.2023.10266645.

[16] "GÉANT network," last accessed 10 May 2024. [Online]. Available: https://network.geant.org/gn4-3n/

[17] "The Internet Topology Zoo," last accessed 14 February 2024. [Online]. Available: http://www.topology-zoo.org/toolset.html

[18] ESnet, "iPerf - The ultimate speed test tool for TCP, UDP and SCTP," last accessed 10 May 2024. [Online]. Available: https://www.iperf.fr/

[19] "Mininet," last accessed 10 May 2024. [Online]. Available: https://mininet.org/

[20] Linux Foundation, "Open vSwitch," last accessed 10 May 2024. [Online]. Available: https://www.openvswitch.org/

[21] "Ryu SDN Framework," last accessed 10 May 2024. [Online]. Available: https://ryu-sdn.org/

[22] "Keras," last accessed 10 May 2024. [Online]. Available: https://keras.io/

electronics

Article

Hierarchical Traffic Engineering in 3D Networks Using QoS-Aware Graph-Based Deep Reinforcement Learning

Robert Kołakowski [1,2,*], Lechosław Tomaszewski [1], Rafał Tępiński [1] and Sławomir Kukliński [2]

[1] Orange Polska, Orange Innovation Poland, 02-326 Warsaw, Poland; lechoslaw.tomaszewski@orange.com (L.T.); rafal.tepinski@orange.com (R.T.)
[2] Division of Cybersecurity, Institute of Telecommunications, Faculty of Electronics and Information Technology, Warsaw University of Technology, 00-637 Warsaw, Poland; slawomir.kuklinski@pw.edu.pl
* Correspondence: robert.kolakowski@orange.com

Abstract: Ubiquitous connectivity is envisioned through the integration of terrestrial (TNs) and non-terrestrial networks (NTNs). However, NTNs face multiple routing and Quality of Service (QoS) provisioning challenges due to the mobility of network nodes. Distributed Software-Defined Networking (SDN) combined with Multi-Agent Deep Reinforcement Learning (MADRL) is widely used to introduce programmability and intelligent Traffic Engineering (TE) in TNs, yet applying DRL to NTNs is hindered by frequently changing state sizes, model scalability, and coordination issues. This paper introduces 3DQR, a novel TE framework that combines hierarchical multi-controller SDN, hierarchical MADRL based on Graph Neural Networks (GNNs), and network topology predictions for QoS path provisioning, effective load distribution, and flow rejection minimisation in future 3D networks. To enhance SDN scalability, introduced are metrics and path operations abstractions to facilitate domain agents coordination by the global agent. To the best of the authors' knowledge, 3DQR is the first routing scheme to integrate MADRL and GNNs for optimising centralised routing and path allocation in SDN-based 3D mobile networks. The evaluations show up to a 14% reduction in flow rejection rate, a 50% improvement in traffic distribution, and effective QoS class prioritisation compared to baseline techniques. 3DQR also exhibits strong transfer capabilities, giving consistent performance gains in previously unseen environments.

Keywords: 6G; AI; MADRL; SDN; traffic engineering; GNN; user plane; 3D networks

Academic Editor: Christos J. Bouras

Received: 1 February 2025
Revised: 26 February 2025
Accepted: 3 March 2025
Published: 6 March 2025

Citation: Kołakowski, R.; Tomaszewski, L.; Tępiński, R.; Kukliński, S. Hierarchical Traffic Engineering in 3D Networks Using QoS-Aware Graph-Based Deep Reinforcement Learning. *Electronics* **2025**, *14*, 1045. https://doi.org/10.3390/electronics14051045

1. Introduction

One of the key targets of the 6th Generation (6G) mobile network is ubiquitous coverage and broadband network access worldwide [1]. While extending the terrestrial infrastructure in remote locations is often infeasible due to financial, physical, or legal reasons, network omnipresence is expected to be achieved via the integration of Terrestrial Networks (TNs) and Non-Terrestrial Networks (NTNs) [2–4], including both aerial and satellite systems. Nowadays, carrier-grade aerial networks are largely limited and correspond mainly to High Altitude Platform Systems (HAPSs) or ad hoc networks based on Unmanned Aerial Vehicles (UAVs) (set up dynamically during natural disasters, mass events, etc.). Currently, the primary interest lies in the integration of TNs with already operating satellite systems with a focus on Low Earth Orbit (LEO) systems due to acceptable latency (30–50 ms) [5].

The LEO networks, however, introduce several challenges caused by the fast mobility of infrastructure causing frequent topology changes [6] and severe capacity constraints—both

Inter-Satellite Links (ISLs) and Feeder Links (FLs). To maintain service continuity and economise NTN resources, it is necessary to adopt intelligent Traffic Engineering (TE) for Quality of Service (QoS) routing, End-to-End (E2E) cross-layer operations, and network-wide load balancing. The latter is critical to mitigating the impact of the disappearing links and related distribution of rerouted traffic across multiple links (not necessarily the closest ones as in shortest path routing). In TNs, Deep Reinforcement Learning (DRL) is a common candidate method for TE due to high performance and adaptability [7,8]. Its application in NTNs is, however, problematic due to the strong dependence of the learned policies on the problem setup. Time-variant environments complicate the exploitation of traditional "deep" architectures and degrade DRL's models convergence. Moreover, DRL faces observability and scalability issues, and its usage in distributed and decentralised environments is still being heavily researched [9–12].

Software Defined Networking (SDN) is often perceived as a vital concept to introduce flexibility into the network and facilitate cross-layer integration [13]. The logically centralised software-driven control, support for TE, and capabilities to leverage contextual information are essential to optimise networking in the evolving NTN segment. The usage of SDN in carrier-grade networks is, however, still limited due to scalability, interoperability [14], and SDN Controller (SDNC) placement issues [15]. While distributed architectures address the above partially, they complicate E2E TE and E2E QoS management and require SDNCs coordination.

In this paper, based on the authors' previous work [16], proposed is a novel TE approach called 3D QoS-Aware Routing (3DQR) for optimising routing and path allocation in future mobile Three Dimension (3D) networks. The key contributions of this paper are as follows:

- Considering the heterogenous and distributed character of emerging 6th Generation (6G) networks, introduced is a TE framework based on distributed (multi-controller) hierarchical SDN and Multi-Agent Deep Reinforcement Learning (MADRL), which enables flexible global and per-segment optimisation of 3D networks (i.e., comprising terrestrial, aerial, and satellite segments). The proposed framework improves load distribution and flow acceptance rate while considering QoS requirements of individual flows and traffic priorities. Moreover, 3DQR tries to minimise the number of broken paths and path reconfigurations to reduce session disruptions and SDN Control Plane (CP) overhead to improve SDN scalability.
- A hierarchical MADRL routing and path allocation strategy is developed, which involves intelligent DRL-Graph Neural Network (GNN) TE agents leveraging message passing and attention mechanisms as well as network topology predictions to improve the reasoning of agents and adaptability to frequently changing 3D network topology.
- Based on evaluation, an over 13% reduction in flow rejection rate and a 50% improved load distribution compared to baseline routing methods are demonstrated. Also, strong generalisation and transfer capabilities of 3DQR agents are demonstrated, which enable their effective exploitation in previously unseen topologies.

The paper is structured as follows. Section 2 outlines key routing challenges in 3D networks. Section 3 describes related works on TE focusing on SDN-based frameworks, QoS enforcement, and DRL applications. Section 4 presents the 3DQR architecture, principles, and E2E path establishment workflow. Section 5 presents the details of the 3DQR algorithm, including mathematical formulation, DRL problem setup, agent architecture, and E2E operation. Section 6 describes the evaluation and obtained results. Section 7 outlines open issues and potential future works. Finally, Section 8 summarises and concludes the paper.

2. Routing Challenges in TN-NTN Mobile Networks

Integration of terrestrial, aerial, and satellite network segments will enable an omnipresent network. Fast mobility of the satellite part poses several challenges and issues regarding the organisation of routing and its optimisation. The key challenges to be tackled to enable unified 3D networks include the following [3,5,17]:

(C1) **Routing convergence**—applying traditional Internet Protocol (IP) routing schemes, e.g., Open Shortest Path First (OSPF), is problematic in NTNs due to mobility of network nodes. Frequent reconfigurations of connections between nodes require continuous updates of routing tables and link costs to maintain up-to-date information on the network state within the nodes. The changes occur repeatedly, causing almost constant updates, which impacts the convergence and leads to unstable routing and large signalling overhead [18]. Adoption of SDN to provide dynamic and flexible user traffic steering is a common solution [18]. However, the original SDN concept lacks CP scalability, so its wide-scale deployments are problematic. Distributed SDN architectures allow for mitigating SDNC overload at the cost of complexity—E2E functionality requires the development of coordination mechanisms across multiple SDNCs. Also, in wide-scale deployments, SDNCs placement for optimal network control needs careful consideration (in terms of both intra- and cross-layer CP latency, network observability (obtaining reliable network monitoring information), and recovery).

(C2) **Temporal and predictive routing**—topology changes caused by the mobile NTN nodes can lead to broken links or changes of link properties (e.g., bandwidth drops due to partial occlusion). Addressing these issues will require frequent rerouting of flows and path updates on a topology change, leading to extensive signalling traffic. To calculate viable routes with increased durability and mitigate the above issues, link availability prediction and mechanisms for fast updates of paths (via low-latency monitoring, TE metrics adjustments, etc.) are vital. These are required to support time-scheduled routing schemes and enhancements by contextual information (e.g., satellite orbits, air interfaces alignment, object occlusion, etc.).

(C3) **Resilience**—the disappearing links and broken network paths can lead to severe Service Level Agreement (SLA) violations. Hence, increasing resilience to minimise the impact of communication cutoffs is a core requirement to enable QoS-driven services. Potential solutions include multi-path and/or node/edge-disjoint routing [19], in-switch buffering mechanisms in case of access node isolation (i.e., store-and-forward) [20,21], or predictive routing schemes.

(C4) **Optimisation**—conventional TE algorithms—e.g., Multi-Protocol Label Switching–Traffic Engineering (MPLS-TE)—cannot be used effectively in NTNs due to long convergence time. The emerging routing and TE methods will need to consider both QoS constraints and effective traffic distribution to handle limited ISL capacity. Moreover, the synchronisation of network state information across TE databases and its unified representation in NTNs becomes crucial to facilitate E2E control and Artificial Intelligence (AI)-driven optimisation. To achieve the latter, the TE algorithms need to be able to extract the information from the nodes and link relationships rather than the fixed graph structure to avoid overly complex and monolithic models.

(C5) **Asset heterogeneity**—future mobile networks are expected to combine heterogeneous systems with diversified service capabilities and properties (radio interfaces, protocol stack, etc.), i.e., the *Network of Networks* [22]. To enable E2E services in 3D networks, it is essential to develop dynamic QoS management and coordination mechanisms to provide paths satisfying E2E QoS regimes. Also, to handle the rising complexity, the network management will need to embed automated and intelligent TE mechanisms enabling cross-domain cooperation, knowledge transfer to new segments, and seamless operation.

3. Related Work

In recent years, many works addressing the problems of intelligent TE in TNs, NTNs, and integrated 3D networks have been published. Here, in focus are key routing algorithms and architectural frameworks that target optimisation under QoS constraints.

Organisation of routing is one of the key issues regarding carrier-grade NTNs. In traditional TNs, routers calculate routing tables using a routing information database established based on network topology and link states. Routing updates require exchanging topology information across network devices. It leads to significant overhead and slow convergence [23], causing the inapplicability of traditional approaches in NTN scenarios due to limited resources and frequent topology changes. Therefore, academia and industry started to develop routing algorithms adapted to the NTN specificity [24]. A time–topology routing algorithm for 3D networks is presented in [25], which outperforms existing message-based routing methods in terms of latency and delivery rate. The method considers node mobility and contextual information (weather, propagation models, etc.) to obtain a time-based inter-node visibility and link information to construct the network topology for a certain time frame and generate routes providing stable communication. A QoS-aware routing minimising congestion in SDN-based NTN systems is proposed in [26]; the method assigns scores to network links based on selected QoS parameters (e.g., utilisation, congestion degree, delay). On that basis, it finds shortest paths using a modified Bresenham's algorithm. A compass time–space model for the scalable representation of mass-scale NTN systems and the QoS routing method is introduced in [27]. The algorithm yields QoS performance and reliability improvements compared to baseline methods. A cooperative data downloading routing scheme for LEO satellites, which uses ISL to schedule feeder link bandwidth resources, is proposed in [28]. The routing algorithm considers the limited ISL bandwidth and utilises linear programming for solving the route selection problem. Another popular NTN routing approach to provide resilience and stable QoS involves exploiting multi-path routing. A Multi-Path routing algorithm with Joint Optimisation of Load balancing (MPJOL) [29] proposes clustering of LEO satellites and selection of cluster head nodes (considering connectivity and load capacity) to reduce the network scale, and calculating k-shortest paths between the source and destination clusters. The method improves transmission efficiency and quality while reducing the route discovery overheads. The Multi-Path QoS Routing (MPIR) approach given in [30], which targets QoS improvements, proposes an algorithm to generate QoS routes using a genetic algorithm fueled by simulated annealing to handle population diversity and improve convergence. While increasing resilience, multi-path routing is, however, inefficient in large-scale networks [29] due to traffic multiplication. Another common dynamic routing approach is to embed the information about the routes within the packets and exploit Source Routing (SR) [31] or Segment Routing [32]. These solutions, however, increase the Data Plane (DP) traffic volume due to additional headers attached to each conveyed packet. Also, SR-based methods require continuous updates of preset paths in gateway nodes, which can result in significant operational overhead.

With the recent onset of AI and Machine Learning (ML), more researchers have directed their attention towards using AI-based routing in complex environments such as Space-Air-Ground Integrated Networks (SAGINs) [33]. AI-driven routing (commonly DRL-based) typically aims to solve specific issues of conventional routing methods in dynamic topologies, i.e., convergence, QoS, and resource efficiency. A two-hops state-aware Double Deep Q-Network (DDQN)-based routing strategy for LEO networks called DRL-THSA [34] assumes that a node needs only link-state information of neighbouring nodes up to two hops away for the prediction of the next hop. The approach shows solid performance regarding the E2E delay, throughput, packet drop rate, and traffic distribution. A DRL-based satellite routing method solving multiple path requests at once to minimise latency is proposed in [35]. The centralised DRL, however, does not offer high performance, especially in wide-scale network deployments in which components are spatially distant due to network observability issues. Therefore, to optimise complex and distributed networks, such as NTN and 3D networks, a common approach is to use Multi-Agent Deep Reinforcement Learning (MADRL) methods featuring flat or hierarchical structures. A two-stage MADRL framework has been introduced for mobile edge computing [11], which employs high-level agents for the optimisation of transmitting power and duration of wireless transfer power, and low-level agents to improve offloading decisions of wireless devices, time allocation for offloading, and used computing resources. A distributed routing algorithm based on the Transformer-MIX architecture and MADRL, which optimises E2E latency and load balancing, has been proposed [9]. In the concept, the LEO satellites can make individual next-hop decisions for the packets in real-time while participating in the centralised training scheme focusing on load balancing. A MADRL approach, which uses agents enhanced with self-attention mechanisms to first better understand the semantics of network state and then optimise routing in LEO networks, has been proposed [10]. The method enables lowering E2E latency and improves average throughput compared to other distributed routing methods.

Another emerging trend in AI-driven routing is using GNNs for more efficient network state representation and predictions to fuel AI reasoning and performance. A method using GNNs is proposed to handle routing strategies devoted to finding the shortest path and maximising the minimum allocated bandwidth for improved load distribution across network links in [36]. Another approach uses GNNs to predict packet delay distribution and loss and uses these predictions for routing optimisation and network planning [37]. A Graph-Aware Deep Learning model is proposed [38] to maintain precise network measurements along with current positions and destinations of forwarding tasks. The Contact Graph Routing algorithm is proposed with a mix of reinforcement and Bayesian learning for delay-tolerant networks in [39]. A topology-aware GNN framework for link prediction highlighting the trade-offs between the algorithm precision and speed, which can also be used for routing, is introduced in [40]. Another GNN-based routing method for terrestrial optical networks combining Message Passing Neural Network (MPNN) and DRL is proposed in [41]. The method yields good performance and generalises well on unseen topologies. A graph-based MADRL approach called GraphPR has been proposed [12] to optimise routing in satellite networks with partial observability. The method employs Graph Attention Networks to transform satellite states (e.g., location, queue, neighbour states), which are later processed by MADRL to make routing decisions. Due to the increasing popularity of GNNs, a benchmark system aimed at accelerating the development of quality GNNs has been developed and described [42].

Clearly, 3D networks will comprise multi-tier, multi-dimensional and heterogeneous network architectures with different capabilities and properties that will complexify QoS-driven communication [43,44]. The most popular State of the Art (SotA) approach is the adoption of distributed architectures with extensions implementing E2E functionality via cross-domain coordination. A hybrid hierarchical and multi-controller SDN architecture to tackle resource management and minimise SDNC signalling overhead in heterogeneous 3D networks is proposed [43]. The SDNCs, handling ground, aerial, and satellite layers, caters towards layer-specific QoS requirements and routing, while the E2E routes are combined from the intra-layer ones in the service composition layer. Service-Customised Network (SCN) for Immersive Media in SAGINs is studied in [44]. The concept proposes hierarchical SDN deployment for increased management capabilities and usage of edge servers for reduced end-to-end latency. It also introduces a DRL-enhanced multi-path Transport Control Protocol (TCP) routing strategy, which uses network state as an input and user Quality of Experience as feedback. Another hierarchical multi-tier SDN-based framework for integrated vehicular networks is proposed in [45], which introduces multi-level SDNCs (local, regional, national, and global domains) that communicate to enable coordination across segments and optimise network services.

Summing up, SotA approaches strongly lean towards exploiting SDN for 3D network architectures. With increasing complexity, heterogeneity, and dynamics comes the need for efficient and tailored routing solutions. AI/ML-based methods, especially GNN-based ones, outperform conventional routing algorithms and enable good generalisation in complex environments. The latter is achieved thanks to the GNN ability to preserve the graph structure of data (compared to Deep Neural Network (DNN) architectures) and perform embedding operations. The node/edge feature embedding enables encoding of spatial and relational dependencies of the input data into low-dimensional space, taking into account not only the feature values but also the connectivity between nodes. In the context of DRL methods, GNNs bring multiple benefits that include learning meaningful state representations, improving output policy quality, reasoning, and model generalisation. As GNNs can process different-sized graphs, compared to traditional fixed-size DRL, they enable operation in frequently changing environments, such as NTNs and LEO, eliminating the need for multiple agents or complex architectures supporting state representation processing into a fixed-size input (as it would be in the DNN case).

While deeply investigated in SotA, the usage of GNNs and DRL for routing, and their benefits regarding performance, QoS provisioning, or load balancing, as well as the SDN CP overhead, scalability aspects, and operator, profits are usually overlooked. Moreover, there is a general lack of methods for AI-driven 3D routing that consider the performance of individual network segments in addition to the E2E one. Although numerous distributed routing approaches are proposed (also GNN-based ones), these usually focus on selecting the next hop, giving limited abilities in terms of global-level network optimisation. This gap is filled by proposing an intelligent QoS-aware routing approach for SDN-based 3D networks that optimises load distribution, considers network operator priorities, and limits the impact on the SDN CP caused by frequent rerouting.

4. 3D QoS-Aware Routing (3DQR)

4.1. Concept Principles

Infrastructure mobility and the mitigation of its effects are a core challenge for TN and NTN integration. Introducing User Plane (UP) flexibility and mechanisms for optimal routing and QoS path allocation is vital for provisioning stable network services to the end customers worldwide (cf. Section 2). The hereby proposed 3DQR concept aims to address this challenge by combining the powers of SDN, DRL, GNNs, and topology prediction. In 3DQR, a network comprised of a multi-stratum (terrestrial, aerial, space) and multi-domain (multi-controller) hierarchical SDN architecture is considered, as shown in Figure 1.

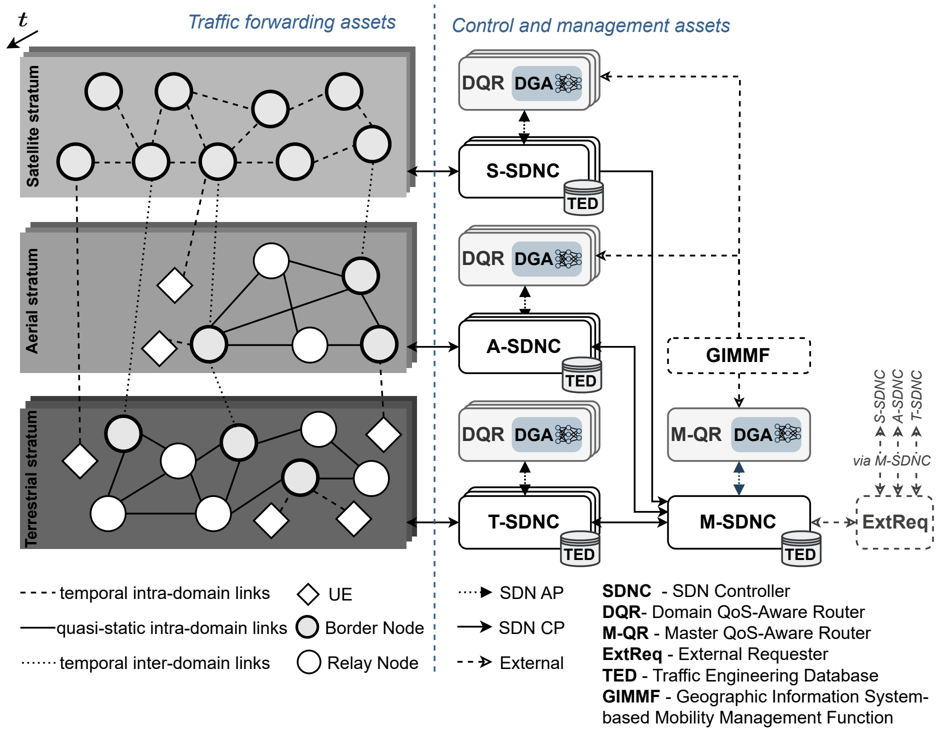


Figure 1. Overall view of 3DQR concept and interactions across components.

The network assets within strata (terrestrial, aerial, and satellite) form SDN domains (one or more per stratum). Each domain is controlled by a dedicated SDNC (one per domain), i.e., Terrestrial SDNCs (T-SDNCs), Aerial SDNCs (A-SDNCs), and Satellite SDNCs (S-SDNCs). SDNCs implement domain-specific CP protocols (e.g., OpenFlow (OF) [46]), enabling full control of the traffic flows within a domain (i.e., of SDN switches) and monitoring suite for network state information acquisition. Each SDNC is accompanied by a dedicated Domain QoS-aware Router (DQR) with embedded DRL-GNN Agent (DGA) (further described below)—TE application is responsible for the calculation of optimal routes for all of the flows traversing the domain and load distribution. The E2E functionality is implemented by the logically centralised and physically distributed global-level Main SDNC (M-SDNC) acting as the umbrella controller for the underlying set of SDNCs and assisted by Master QR (M-QR)—DQR equivalent handling cross-domain routing and load balancing. M-SDNC operates using the overlay network view of the network that refers to the following:

- Monitoring—in spatially distributed networks, centralised SDNC suffers from CP link latency, which leads to obtaining obsolete monitoring data. The hierarchical approach partially addresses this issue, as SDNCs can be deployed in close proximity of switches. Moreover, growing network size increases the monitoring data volume dramatically. Therefore, instead of link-level metrics, SDNCs calculate the parameters of overlay links between domain ingress/egress nodes denoted as Border Nodess (BNs) (access nodes, gateways, satellites, etc.; cf. Figure 1). This reduces the monitoring traffic while conveying information about domains' abilities to serve new flows (cf. Section 5.1).
- CP operations—M-SDNC has a view limited to BNs, overlay links within domains, and interconnection links to domains. Hence, M-SDNC arranges the End-to-End Path (E2EP) enforcement by instructing individual SDNCs to allocate Intra-Domain Path (IDP) between BN pairs (i.e., a path composed of two BNs and intermediary relay nodes). As inter-domain links are either physical or overlay links between BNs of

two domains, installing relevant flow entries in the BNs is equivalent to establishing the inter-domain path. The full E2EP allocation procedure is described in Section 4.2.

The topology information, monitoring metrics, and flow allocation history are stored in dedicated Traffic Engineering Databases (TEDs)—local and E2E—and exposed to TE applications on demand. TED implements the data retention strategies based on the requirements of operating TE mechanisms (especially DRL-GNN agents described further in this section). The details, however, are out of the scope of this paper.

The ultimate goal of 3DQR is to perform optimal routing and E2E path allocation in the integrated 3D network, considering three major aspects: network-level performance (measured by the degree of traffic distribution over the network links), flow QoS constraints, and flow importance (determined arbitrarily by the network operator based, e.g., on service priority or associated profits). These objectives are achieved by the federation of DQRs and a centralised M-QR, which provide optimal paths on SDNCs/M-SDNC requests (domain-level and E2E). Unlike traditional IP networking, where a router is a DP- and CP-integrating device that makes local packet routing decisions, in 3DQR, the router is an Application Plane (AP) function interacting with the SDNC's NorthBound Interfaces (NBIs). The selection of the most profitable paths is made by the intelligent DGA implementing the DDQN algorithm [47] and embedded in each DQR/M-QR. The data flow within DQR and DGA is shown in Figure 2.

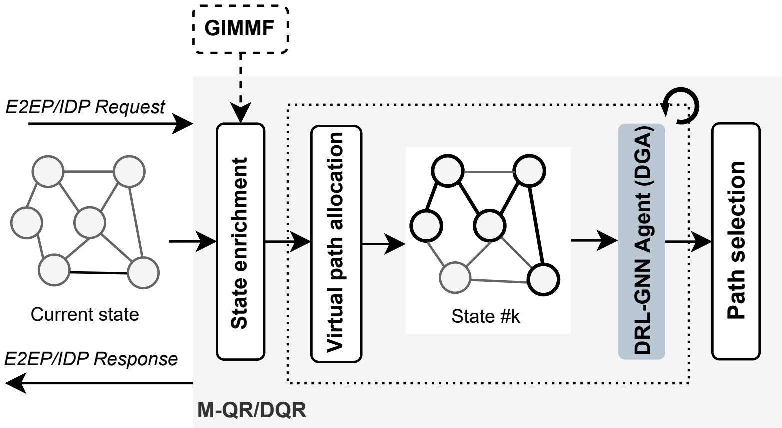


Figure 2. M-QR/QR and DGA architecture and data flow.

Upon the arrival of a path request (cf. Section 4.2), DQR collects the most recent network state information (e.g., OF-based statistics—cf. Section 5.2) and extends its node/edge feature vectors by additional node/link state information such as node centrality or nodes/links expiry information (probability of nodes' reachability or existence of ISLs and gateway links in the considered time-frame). The latter is provided by a dedicated Geographic Information System-based Mobility Management Function (GIMMF) entity, which estimates NTN topology (active ISLs, FLs, etc.) and calculates the aggregated persistent graph within a time frame (i.e., containing only the set of nodes and links that is not expected to change), using the orbital/trajectory parameters of nodes. DQR computes then k shortest paths using delay as the metric. For each candidate path, DQR/M-QR performs virtual path allocation using flow's QoS class requirements and feeds the outcome state to the DGA to obtain the value of the allocation. The path is selected based on the highest Q-value across k virtual path allocations. Additionally, to tackle the issue of variable size of network states in NTN, DGAs exploit the GNN architecture to extract the information about the relationship between nodes and links to enable effective routing regardless of topology shape and size (cf. Section 5.3).

It must be noted that SDNCs and M-SDNC are solely responsible for verifying the feasibility of a path and its enforcement while the reasoning is left to DQRs. Moreover, as each satellite/aerial node can act as an access point capable of serving the User Equipment (UE), they are considered BNs. Finally, to enable straightforward integration of the concept with 3rd Generation Partnership Project (3GPP) networks, QoS classes are aligned with the 3GPP definition, i.e., 5G QoS Identifier (5QI) [48].

4.2. E2E Routing and Path Allocation Approach

The paths are requested by the External Requester (ExtReq) entity (#1) such as UP entities, Management and Orchestration (MANO), M-SDNC applications, etc. M-SDNC, which constitute a single entry point to the framework by exposing path allocation Application Programming Interfaces (APIs). The E2EP allocation process is shown in Figure 3. The path allocation process also includes the reservation of resources for the traffic flow with respect to the flow's QoS class.

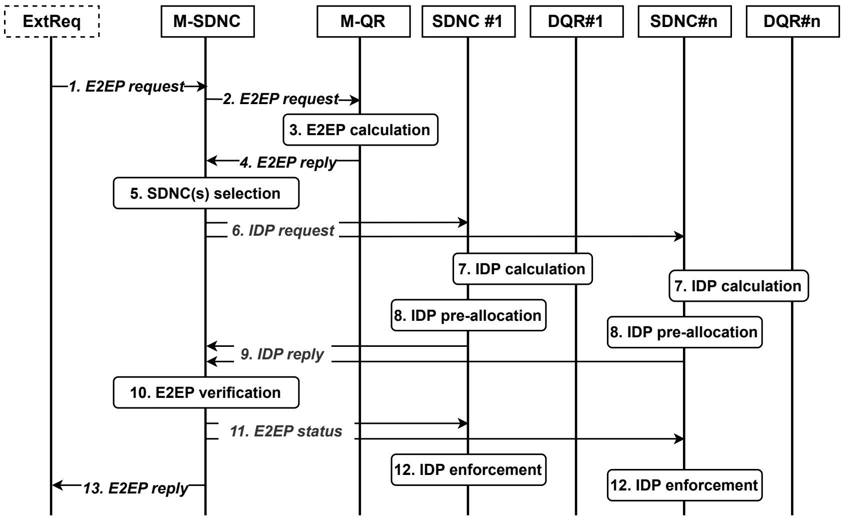


Figure 3. Path allocation in 3DQR concept.

First, M-SDNC sends the E2EP request containing flow-specific data (source node, target node, and QoS class) to M-QR (#2). M-QR calculates the E2EP (#3), which consists of BNs and overlay links visible at the M-SDNC level (#4). After verification of E2E path allocation feasibility, M-SDNC splits the path into segments based on the BNs domain membership (#5). For each segment, M-SDNC requests IDP calculation from the SDNC-DQR pairs (#6, #7). Then, IDPs' feasibility is verified by SDNCs, which ends with the pre-allocation of resources of the flow (#8). Finally, SDNCs notify M-SDNC about the successful pre-allocation of resources for the IDPs (#9), including the QoS parameters of the selected paths (e.g., delay, expected error rate, etc.). After receiving a positive notification from each domain, M-SDNC verifies if connected IDPs satisfy the E2E QoS requirements (#10). If so, M-SDNC notifies respective SDNCs about successful E2EP verification (#11) to finalise enforcement (#12), and notifies ExtReq about successful allocation (#13). If IDP allocation is impossible, e.g., due to dynamic changes in the network (nodes mobility, faults, etc.), SDNC notifies M-SDNC about it; then, M-SDNC requests respective SDNCs to revert IDPs allocations. The same approach is taken by M-SDNC if E2EP verification fails.

While the centralised approach increases procedure duration, it ensures E2EPs even in disjoint NTN topology graphs, i.e., missing ISLs, or in a multi-stakeholder environment (e.g., NTN roaming). The inclusion of an M-SDNC overlay view enables routing the traffic via TN or other NTN segments to connect the disjoint parts of topology and attempt the allocation.

5. Algorithm

This section presents the details of the 3DQR algorithm and related mathematical formulations. 3DQR targets routing challenges in 3D networks (cf. Section 2) by carrying out the following:

- Using SDN-based deployment for flexible E2E routing and TE operations [C1, C4];
- Leveraging GIMMF-based predictions to improve DGA reasoning [C2] and to establish a topology containing stable links within a given time frame [C3];
- Providing a novel approach combining DDQN and GNN to (i) provide intelligent routing and path allocation decisions in 3D networks; (ii) enable variable size input; and (iii) embed both network- and flow-level metrics for evaluation of routing decisions [C4];
- Adopting a distributed SDN architecture and modular routers, enabling heterogeneous technologies at the domain level and dynamic attachment of new domains [C1, C5].

The 3DQR operation details are presented in the following sections.

5.1. System Model

The problem of QoS routing and path allocation in an integrated TN-NTN network is modelled as a time-expanded minimum-cost multi-commodity flow. The overall network graph $G(V, E, t)$ is composed of a set of domain sub-graphs, each belonging to a domain described by a unique identifier $\mathfrak{d} \in \mathbb{N}$. Each sub-graph is time-variant $G^{\mathfrak{d}}(V^{\mathfrak{d}}, E^{\mathfrak{d}}, t)$ in the NTN case (space and aerial strata) or time-invariant $G^{\mathfrak{d}}(V^{\mathfrak{d}}, E^{\mathfrak{d}})$ in the TN case. The graphs are constituted by sets of nodes and edges, i.e., $V = [v_1, v_2, \ldots v_j]$ and $E = [e_{12}, e_{13}, \ldots e_{ij}]$, with e_{ij} representing a bidirectional edge between vertices v_i and v_j. Edge properties are described by a total of six features ($F = \{f_{ij} | (v_i, v_j) \in E\} \in \mathbb{R}^{|E| \times m}$, m—dimensionality of features) that include total capacity c_{ij}, total Guaranteed Flow Bit Rate (GFBR)—cf. [48], clause 5.7.2.5—allocated to flows b_{ij}^{GFBR}, total Maximum Flow Bit Rate (MFBR)—cf. [48], clause 5.7.2.5—that can be consumed by flows b_{ij}^{MFBR}, delay d_{ij}, packet error rate δ_{ij}, link utilisation $u_{ij} = b_{ij}^{\text{GFBR}} / c_{ij}$, indicating the fraction of capacity currently allocated to flows, and ζ_i—the vector containing the share of flow types traversing the edge (one value per QoS class identifier). The nodes' features ($X = \{x_i | v_i \in V\} \in \mathbb{R}^{|V| \times m}$) correspond to the ones defined for edges, i.e., including node capacity (cf. Equation (1)), guaranteeing the maximum bandwidth allocated to flows traversing the node (cf. Equations (2) and (3)) and node utilisation (cf. Equation (4)). The edges incident to the vertex v_i are denoted with the $E(v_i)$ operator—i.e., $E(v_i) = \{e \in E : \exists v_j \in V, e = (v_i, v_j) \in E\}$. The delay and packet error rate are excluded from node features as they are both usually included in the link parameters due to measurement strategies that span over links and connecting switches egress/ingress ports [49]. The time consumed by switches on flow matching and forwarding is negligible and omitted.

$$c_{v_i} = \sum_{e_{ij} \in E(v_i)} c_{ij} \tag{1}$$

$$b_{v_i}^{\text{GFBR}} = \sum_{e_{ij} \in E(v_i)} b_{ij}^{\text{GFBR}} \tag{2}$$

$$b_{v_i}^{\text{MFBR}} = \sum_{e_{ij} \in E(v_i)} b_{ij}^{\text{MFBR}} \tag{3}$$

$$u_{v_i} = \frac{b_{v_i}^{\mathrm{GFBR}}}{c_{v_i}} \tag{4}$$

The edge metrics are used by SDNCs to calculate the overlay view for M-SDNC/M-QR. The corresponding overlay metrics and topology are denoted with a superscript ψ. The metrics transformation takes into account only the BNs of the domain ($V^B \subset V$) so that M-SDNC sees the viable allocations between domain gateways. The bandwidth-related metrics between a pair of BNs, $v_i, v_j \in V^B$, are calculated using maximum flow (Ford–Fulkerson, Edmonds–Karp, Push–Relabel, etc.) and shortest path (e.g., Dijkstra) algorithms, denoted as $maxflow$ and SP, extending the concept [16]. These include the following:

- Total capacity c_{ij}^{ψ} measured with no traffic between v_i and v_2:

$$c_{ij}^{\psi} = maxflow(v_i, v_j) \tag{5}$$

- Aggregate of GFBRs allocated to flows $b_{ij}^{\psi,GFBR}$:

$$b_{ij}^{\psi,GFBR} = c_{ij}^{\psi} - maxflow(v_i, v_j) \tag{6}$$

- Peak aggregated bandwidth (aggregate of MFBRs) $b_{ij}^{\psi,MFBR}$ that can be consumed by allocated flows:

$$b_{ij}^{\psi,MFBR} = \sum_{(u,w)\in SP(v_i,v_j)} b_{uw}^{MFBR} \tag{7}$$

- Utilisation of overlay link u_{ij}^{ψ} between v_i and v_j:

$$u_{ij}^{\psi} = (c_{ij}^{\psi} - b_{ij}^{\psi,GFBR)} / c_{ij}^{\psi} \tag{8}$$

The delay and packet error rate are approximated by calculating the shortest path obtained above and aggregation/multiplication over the edges belonging to the path:

$$d_{ij}^{\psi} = \sum_{(u,w)\in SP(v_i,v_j)} d_{uw} \tag{9}$$

$$\delta_{ij}^{\psi} = 1 - \prod_{(u,w)\in SP(v_i,v_j)} (1 - \delta_{uw}) \tag{10}$$

Each traffic flow is modelled by a tuple $f = (v_{src}, v_{dst}, t_0, t_d, q)$, where v_{src}—source, v_{dst}—sink, t_0—arrival time, t_d—duration, and q_i—an SDN-level QoS class. Following the simplified 3GPP 5QI definition, q_i is modelled as a tuple $q_i = (b^{\mathrm{GFBR}}, b^{\mathrm{MFBR}}, d^{\max}, \delta^{\max})$, where q_{i}—a QoS class identifier (common across SDN domains/strata), b^{GFBR}—GFBR (corresponding to the bandwidth reserved by SDNC/M-SDNC), b^{MFBR}—MFBR, d^{max}—Packet Delay Budget (PDB) (cf. [48], clause 5.7.3.4), and δ^{max}—Packet Error Rate (PER) (cf. [48], clause 5.7.3.5). Hereby, each flow is considered as unidirectional, due to commonly different proportions of uplink (UL) and downlink (DL) traffic. In such cases, allocation of two-way exchange requires using different q_i values for UL and DL flows. Each allocation of flow f results in either acceptance or rejection decision $f_d \in 1, 0$.

5.2. DRL Problem Setup

The multi-controller hierarchical SDN can be treated as the combination of standard Reinforcement Learning (RL) settings, i.e., a set of stochastic environments—each modelled as Markov Decision Process (MDP) represented by tuples $M = (S; A; T; R)$, where S denotes states, A actions that can be taken by the agent, T the transition function, and R the reward function. As the environment details are unknown, including transition probability between states and dynamics, the dynamic programming methods [50] cannot effectively solve the given MDP. Moreover, the network environment is a non-linear system (due to complex traffic patterns, bursty behaviour, in-DP buffering, etc.), making the exploitation of well-established methods inapplicable [51]. Hence, the proposed DQR algorithm utilises a model-free DRL due to its suitability for operation in environments in unknown or not-well-understood dynamics.

State: For each arriving flow allocation request, the DRL agent is to evaluate k candidate paths and select the one that maximises the reward. To improve the reasoning of agents, the environment state is obtained as follows. Once the network information is pulled from SDN controllers (i.e., edge/node properties listed in Section 5.1), the graph is extended with more features provided by GIMMF and calculated internally by the router (i.e., state enrichment phase). Additional features include the edge/node expiry $\xi_{ij}, \xi_i \in \langle 0, 1 \rangle$ (probability of node/edge activity in the defined time horizon: 0 for active, 1 for inactive), $\chi(v) \in \mathbb{N}$ (node type, e.g., satellite, gateway, relay, etc.), and $deg(v_i)$ (node centrality). The obtained graph constitutes the environment state $s_t^{\eth} = (G^{\eth}, X^{\eth}, F^{\eth})$ in the DQR and $s_t^{\psi} = (G^{\psi}, X^{\psi}, F^{\psi})$ in the M-QR case, where (X, F) are node and edge features:

$$\begin{aligned} F^{\eth} &= \{f_{ij}^{\eth} = (c_{ij}, d_{ij}, \delta_{ij}, b_{ij}^{\text{GFBR}}, b_{ij}^{\text{MFBR}}, u_{ij}, \zeta_{ij}, \xi_{ij}) | (i,j) \in E^{\eth}\} \\ X^{\eth} &= \{x_i^{\eth} = (c_i, b_i^{\text{GFBR}}, b_i^{\text{MFBR}}, u_i, \xi_i, deg(v_i), \chi(v_i)) | v_i \in V^{\eth}\} \\ F^{\psi} &= \{f_{ij}^{\psi} = (c_{ij}^{\psi}, d_{ij}^{\psi}, \delta_{ij}^{\psi}, b_{ij}^{\psi,\text{GFBR}}, b_{ij}^{\psi,\text{MFBR}}, u_{ij}^{\psi}, \zeta_{ij}, \xi_{ij}) | (i,j) \in E^{\psi}\} \\ X^{\psi} &= \{x_i^{\psi} = (c_i, b_i^{\text{GFBR}}, b_i^{\text{MFBR}}, u_i, \xi_i, deg(v_i), \chi(v_i)) | v_i \in V^{\psi}\} \end{aligned} \tag{11}$$

Before feeding the above state into the DRL agent, a virtual path allocation of the flow for each of the candidate paths is performed (cf. Section 4 and Figure 2 for details). To improve the reasoning of the agent, in addition to the features listed above, flags indicating if the node/edge is a part of the path are included—$v^P, e^P \in 0, 1$. Hence, the state evaluated by DGA for k-th allocation is defined as $s_t^{\eth,k} = (G^{\eth}, X^{\eth,k}, F^{\eth,k})$ for domain graphs, and $s_t^{\psi,k} = (G^{\psi}, X^{\psi,k}, F^{\psi,k})$ for the overlay graph:

$$\begin{aligned} F^{\eth,k} &= \{f_{ij}^{\eth,k} = (c_{ij}, d_{ij}, \delta_{ij}, b_{ij}^{k,\text{GFBR}}, b_{ij}^{k,\text{MFBR}}, u_{ij}^k, \zeta_{ij}, \xi_{ij}, e_{ij}^P) | (i,j) \in E^{\eth}\} \\ X^{\eth,k} &= \{x_i^{\eth,k} = (c_i, b_i^{k,\text{GFBR}}, b_i^{k,\text{MFBR}}, u_i^k, \xi_i, deg(v_i), v^P, \chi(v_i)) | v_i \in V^{\eth}\} \\ F^{\psi,k} &= \{f_{ij}^{\psi,k} = (c_{ij}^{\psi}, d_{ij}^{\psi}, \delta_{ij}^{\psi}, b_{ij}^{\psi,k,\text{GFBR}}, b_{ij}^{\psi,k,\text{MFBR}}, u_{ij}^{\psi,k}, \zeta_{ij}, \xi_{ij}, e_{ij}^P) | (i,j) \in E^{\psi}\} \\ X^{\psi,k} &= \{x_i^{\psi,k} = (c_i, b_i^{k,\text{GFBR}}, b_i^{k,\text{MFBR}}, u_i^k, \xi_i, deg(v_i), v^P, \chi(v_i)) | v_i \in V^{\psi}\} \end{aligned} \tag{12}$$

Action: The action a_t is defined as the path (a list of network nodes) calculated for the flow. IDP is denoted as $a_t^{\eth}$ and E2EP as a_t^{ψ}. As for each flow request, k candidate paths denoted as a_t^k are considered.

Transition: The adopted transition function defines a probability of moving from state s_t to s_{t+1} given the action a_t, $T(s_{t+1}|s_t, a_t)$.

Reward: The reward is given to DGA for each computed path sent to the SDNC for allocation. The reward functions for local and overlay agents are shown in Equation (13) and Equation (14), respectively.

$$r_t = \begin{cases} 1 - std(u) + std\left(\frac{b^{GFBR}}{b^{MFBR}}\right) + RC + QF; & f_d = 1 \\ -QF; & f_d = 0 \end{cases} \tag{13}$$

$$r_t^{\psi} = \begin{cases} 1 - std(u^{\psi}) + RC + QF; & f_d = 1 \\ -QF; & f_d = 0 \end{cases} \tag{14}$$

While the primary goal of 3DQR is to provide QoS paths, the algorithm is expected to consider load balancing (at both the domain and overlay levels), QoS class priorities, and SDN CP overhead. To this end, the proposed reward formula includes multiple terms that guide the agents to carry out the following:

- Distribute path allocations across links to maximise overall throughput—standard deviation of the utilisation of links $std(u)$, and overlay links $std(u^{\psi})$;
- Punish frequent rerouting to conserve SDN CP resources as each flow rerouting requires modification of forwarding rules in switches; this is achieved by using the heuristic Rerouting Cost (RC):

$$RC = \begin{cases} \mathfrak{r};\ \mathfrak{r} \in (-1, 0) & \text{if flow is rerouted} \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

- Prioritise traffic and scale the punishments for allocation failures; therefore, the QoS Factor (QF) heuristic is introduced, which defines the value of each allocated flow based on the QoS identifier q_{id}:

$$QF : q_{id} \mapsto QF(q_{id}) \in (0, 1) \tag{16}$$

- Maximise throughput that can be consumed by flows—if there is remaining capacity in the link, high values of GFBR/MFBR indicate the substantial portion of excess bandwidth that can be consumed by flows; it also encourages the agent to allocate flows with different GFBR/MFBR ratios (commonly associated with different traffic classes) to maximise aggregate the GFBR and decrease the MFBR, allowing the excess bandwidth to be shared across active flows.

5.3. E2E Operation and DGA Architecture

The E2E operation of the 3DQR algorithm is presented in Algorithm 1. 3DQR is based on the model-free and off-policy DRL algorithm called DDQN [47]. The goal of DDQN is to find the close approximation of the optimal action-value function $Q^* : State\ x\ Action \rightarrow \mathbb{R}$ by using a DNN architecture. The obtained function can then be used to construct the policy maximising the acquired rewards, i.e., $\pi^*(s) : argmax\ Q^*(s, a)$. The typical DDQN agent comprises the following:

- Local network—calculating Q-values for actions a_t based on the environment state s_t;
- Target network—stabilising the learning process;
- Replay buffer—storing transitions, actions, and rewards used for training (cf. Figure 4).

The DDQN algorithm is selected as the basis for 3DQR due to its resistance to overestimation bias (at the cost of action-value underestimation in the first steps of operation) [52], which is critical for obtaining a high-quality and generalised policy in a quickly changing environment.

Algorithm 1 3DQR E2E routing and allocation

Input: DDQN params τ, γ, and ϵ, number of candidate paths k, and domain and overlay IDs $\mathfrak{d}, \psi$

Initialise: request queue $\mathfrak{q}_f$ for flows f, for each DQR and M-QR: replay buffer $\mathcal{H}$, primary and target Q-networks Q_θ, Q_{θ^-} with random weights θ and θ^-, virtual path allocation buffer B for tuples (Q-value, post-allocation state, action, path)

for episode $\{1, 2, \ldots, \mathfrak{E}\}$ **do**
    for step t in episode **do**
        $(v_{src}, v_{dst}, f_{q_i}) \leftarrow \mathfrak{q}_f.dequeue()$ ▷ *Figure 3, Step 1*
        $a_{list} \leftarrow [], a_t^{\psi,exp} \leftarrow [], f_d \leftarrow 0$
        $a_t^\psi \leftarrow$ GETPATH$(\psi, v_{src}, v_{dst}, q_i, k)$ ▷ *Figure 3, Steps 2–4*
        if a_t^ψ not empty **then**
            ensure: conditions (17) for a_t^ψ; **else return**
            $a_t^{seg} \leftarrow$ SPLITPATH(a_t^ψ) ▷ *Figure 3, Step 5*
            for $(v_{src}, v_{dst}) \in a_t^{seg}$ **do** ▷ $v_{src}, v_{dst} \in V^B$
                $\mathfrak{d} \leftarrow$ GETDOMAIN(v_{src}, v_{dst})
                $a_t^{\mathfrak{d}} \leftarrow$ GETPATH$(\mathfrak{d}, v_{src}, v_{dst}, q_i, k)$ ▷ *Figure 3, Steps 6–9*
                ensure: conditions (18) for $a_t^{\mathfrak{d}}$; **else return**
                a_{list}.add$(a_t^{\mathfrak{d}})$
            $a_t^{\psi,exp} \leftarrow$ EXPANDPATH(a_t^ψ, a_{list})
            ensure: conditions (19) for $a_t^{\psi,exp}$; **else return** ▷ *Figure 3, Step 10*
            $f_d \leftarrow 1$
            for $\nu \in$ GETDOMAINS(a_t^{seg}) **do**
                $s_t^\nu, a_t^\nu, s_{t+1}^\nu, r_t^\nu \leftarrow$ GETTRANSITION(ν, t)
                $\mathcal{H}^\nu$.add$(s_t^\nu, a_t^\nu, s_{t+1}^\nu, r_t^\nu)$
                every $|t|$ steps: TRAINAGENT(ν)
        return $f_d, a_t^{\psi,exp}$
procedure GETPATH(ν, src, dst, q_i, k): ▷ *graph IDs,* $\nu \in \mathfrak{d} \cup \{\psi\}$
    $s_t \leftarrow$ GETENVIRONMENTSTATE(ν)
    s_t enrichment
    A_t = GETSHORTESTPATHS(G^ν, src, dst, k) ▷ A_t*—candidate paths list*
    for a_t^k in A_t **do**
        s_t^k = VIRTUALPATHALLOC(s_t, p_k, q_i)
        $qval_t^k = Q_\theta^\nu(s_t^k, a_t^k)$
        $B^\nu.store(s_t^k, a_t^k, qval_t^k)$
    $index \leftarrow \text{argmax}_{qval} B^\nu$
    $s_t^\nu, a_t^\nu, qval_t = B^\nu[index]$
    return a_t^ν
procedure SPLITPATH(a_t^ψ):
    subpaths $\leftarrow$ []
    for $i \in 1, 2 \ldots, len(a_t^\psi) - 1$ **do**
        subpaths.add$((a_t^\psi[i], a_t^\psi[i+1]))$
    return: subpaths ▷ $[(v_1, v_2) \ldots (v_i, v_j)]; v_i, v_j \in V^B$
procedure TRAINAGENT(ν, t):
    get sample: $\mathfrak{h}_t^\nu = (s_t^\nu, a_t^\nu, r_t^\nu, s_{t+1}^\nu) \sim \mathcal{H}^\nu$
    $Q^{\nu,*}(s_t^\nu, a_t^\nu) \approx r_t^\nu + \gamma Q_\theta^\nu(s_{t+1}^\nu, argmax_{a^{\nu-}} Q_{\theta^-}^\nu(s_{t+1}^\nu, a^{\nu-})$
    grad. descent step: $((Q^{\nu,*}(s_t^\nu, a_t^\nu) - Q_\theta^\nu(s_t^\nu, a_t^\nu))^2$
    every N steps: $\theta^{\nu^-} \leftarrow \tau * \theta^\nu + (1 - \tau) * \theta^{\nu-}$

The 3DQR algorithm begins with setting up the MADRL environment, which includes initialisation of DQR/M-QR and embedded DGAs (local and target Q-networks, $Q_\theta^{\mathfrak{d}}$ and $Q_{\theta^-}^{\mathfrak{d}}$, with uniformly distributed random weights θ and θ^-, DDQN-specific parameters such as learning rate τ, decay coefficient γ, exploration rate ϵ, the number of candidate paths k, etc.). Initialisation ends with creating the queue $\mathfrak{q}_{\mathfrak{f}}$ inside M-SDNC, which buffers arriving flow allocation requests represented by f. The episode's duration is specified by the number of steps, where each step is a single flow allocation request processed by 3DQR.

Each step begins with retrieving an allocation request (GETFLOWREQUEST) from the queue $\mathfrak{q}_{\mathfrak{f}}$. First, M-QR calculates the rough E2EP containing only the BNs—a_t^{ψ}, using the state visible at the overlay level $s_t^{\psi} = (G^{\psi}, X^{\psi}, F^{\psi})$ (GETPATH—described later in this section). Once calculated, the M-SDNC verifies the feasibility of E2EP considering individual flows' QoS requirements and the limitations of the network infrastructure:

$$\begin{gathered} b_{q_i}^{\text{GFBR}} + b_{ij}^{\psi,\text{GFBR}} \geq c_{ij}^{\psi};\ \forall (i,j) \in a_t^{\psi} \\ \sum_{(i,j)\in a_t^{\psi}} d_{ij}^{\psi} \leq d_{q_i}^{\max} \\ 1 - \prod_{(i,j)\in a_t^{\psi}} (1 - \delta_{ij}^{\psi}) \leq \delta_{q_i}^{max} \end{gathered} \tag{17}$$

If verification fails, M-SDNC delivers the rejection information $f_d = 0$ to the original requester (ExtReq, cf. Figure 3). Otherwise, M-SDNC performs the E2EP split into segments a_t^{seg}(SPLITPATH) composed of BN pairs. For each BN pair in segment $(v_{src}, v_{dst}) \in a_t^{seg}$, M-SDNC sends IDP requests containing the allocation parameters $(v_{src}, v_{dst}, f_{q_i})$ to respective DQRs (GETPATH). The DQR selection is based on the BNs' domain membership (GETDOMAIN). Similarly to the overlay routing and E2EP case, each obtained IDP—$a_t^{\mathfrak{d}}$, undergoes verification by SDNC using constraints shown in Equation (18). The two-level verification aims to eliminate the cases of extreme discrepancies between the overlay "approximate" view of the network and the actual states of domain-level sub-networks, as well as accelerate the rejection process in the case of unavailable QoS-meeting paths.

$$\begin{gathered} b_{q_i}^{\text{GFBR}} + b_{ij}^{\text{GFBR}} \geq c_{ij};\ \forall (i,j) \in a_t^{\mathfrak{d}} \\ \sum_{(i,j)\in a_t^{\mathfrak{d}}} d_{ij} \leq d_{q_i}^{\max} \\ 1 - \prod_{(i,j)\in a_t^{\mathfrak{d}}} (1 - \delta_{ij}) \leq \delta_{q_i}^{max} \end{gathered} \tag{18}$$

The agents are not punished if there is no actual path between the source and destination. The penalty is only for the allocation failures. Hence, if the E2EP does not exist and M-QR returns $f_d = 0$ to M-SDNC, it does not impact the DGA's policy. Moreover, if M-QR selects the path that cannot be allocated due to QoS constraints, the penalty is given to both M-QR and involved DQRs. Once all IDPs are obtained and pre-allocated by SDNCs, the final verification by M-SDNC is conducted by first expanding the E2E path ($a_t^{\psi,exp}$,EXPANDPATH), i.e., substituting overlay path segments with pre-allocated IDPs (actual domain paths $a_t^{\mathfrak{d}}$), and then checking the following conditions:

$$\begin{gathered} \sum_{(i,j)\in a_t^{\psi,exp}} d_{ij} \leq d_{q_i}^{\max} \\ 1 - \prod_{(i,j)\in a_t^{\psi,exp}} (1 - \delta_{ij}) \leq \delta_{q_i} \end{gathered} \tag{19}$$

Finally, if the E2EP satisfies the QoS class requirements, the flow is admitted into the network ($f_d \leftarrow 1$), the E2EP is allocated, and the results are delivered to the original requester. Finally, the transition reward for each domain is calculated (GETTRANSITION), stored in the respective replay buffers $\mathcal{H}$ (i.e, TED), and used for training of the involved DGAs (TRAINAGENT).

To update the current and target Q-networks, the standard DDQN training procedure is followed—i.e., update of the current Q-Network model every $|t|$ steps using the extended Bellman equation (Equation (20)), computing the temporal difference error (calculated using the selected loss function) and its back-propagation to update the model weights.

$$Q^*(s_t, a_t) = r_t + \gamma Q(s_{t+1}, \underset{a}{\text{argmax}}\, Q(s_{t+1}, a; \theta_t^-); \theta_t) \tag{20}$$

Also, a soft update of the Target Q-Network model is performed every N steps [47] with a discount factor τ: $\theta^- \leftarrow \tau * \theta + (1-\tau) * \theta^-$.

The fundamental part of the 3DQR lies in the way of calculating paths, i.e., the GETPATH function that is implemented by M-QR/DQR and a key decision component—DGA. The DQR architecture of both domain- and overlay-level variants is shown in Figure 4.

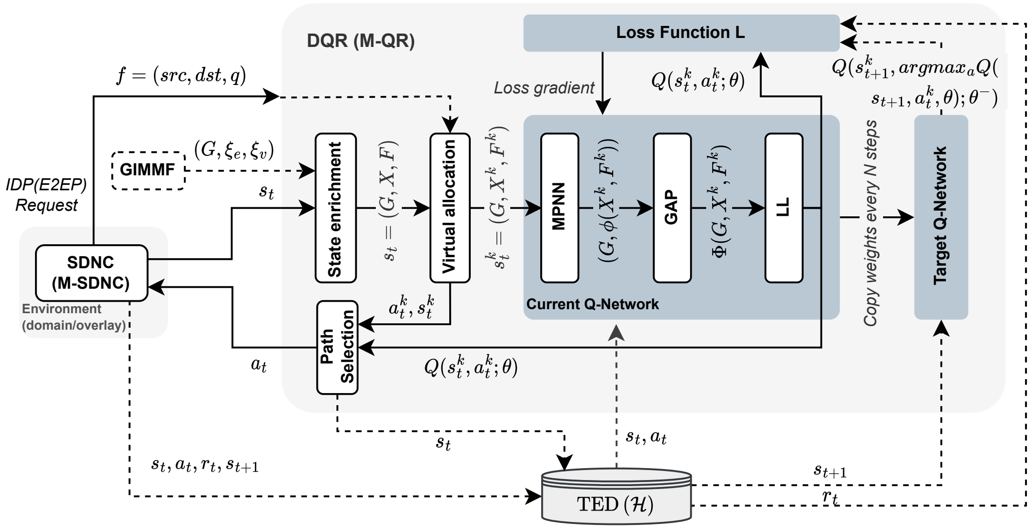


Figure 4. Architecture and interactions of DQR/M-QR and DGA (current and target Q-networks and loss function). Domain and overlay identifiers $\mathfrak{d}, \psi$ are omitted for simplification.

Every IDP/E2EP request issued by SDNC/M-SDNC towards DQR/M-QR includes key flow parameters: source node *src*, destination node *dst*, and QoS class. First, DQR/M-QR gets the current network state s_t (provided by M-SDNC/SDNC; GETENVIRONMENTSTATE) and extends it with additional features including, i.a., node/edge expiry information (cf. Section 5.2). DQR/M-QR also includes the information on persistent NTN graphs, which is delivered by GIMMF (aggregation every 5 s, 10 s, etc.). Once state enrichment is achieved, DQR/M-QR calculates k candidate paths for flow source, destination pairs (f_{src}, f_{dst}) using the Dijkstra shortest path algorithm, and delay as the metric (GETSHORTESTPATHS). For each candidate path, virtual path allocation is performed (VIRTUALPATHALLOC), which involves updating links parameters' (GFBR, MFBR, utilisation, cf. Equation (12)) as if the path allocation succeeded. The obtained state for path k, s_t^k, is later fed to the DGA current policy model $Q_\theta^{\mathfrak{d}}$. For each state, the policy outputs the single Q-value $qval = Q(s_t^k, a_t^k; \theta)$ of the allocation. The tuples of action a_t^k, state s_t^k, and $qval_t^k$ are stored in the buffer $B_v^{\mathfrak{d}}$ to be compared with other virtual path allocations. The final path a_t is selected based on the highest Q-value among the candidate paths and sent back to SDNC/M-SDNC. If no path is available, DQR/M-QR stores the flow metadata to exclude the respective samples from the DGA training process. After the allocation is performed, the respective states in TED are updated with features added by DQR/M-QR (expiry, etc.).

The adopted router architecture design aims to maintain modularity and customisability. The clear separation of concerns between components (modules responsible for state enrichment, virtual path allocation, path selection, and DRL-specific ones—Q-networks and loss function) enables further extensions or tuning based on domain specificity and requirements (e.g., adding additional node/edge features, adopting multi-path routing scheme, using another DRL method or architecture, etc.). Moreover, for easier implementation and integration with SDN frameworks, DQR/M-QR are loosely coupled SDN applications that offer routing services based on provided network data (from SDNC, TED, GIMMF, etc.). Both current and target Q-networks comprise the same neural network architecture. To enable time-variant and size-variant graph input for the agent, a GNN-based approach is adopted. The Q-network architecture is comprised of the following layers:

- Standard MPNN—used to obtain the embeddings ϕ of nodes using both edge and node features. The standard MPNN model is used [53], which defines two phases of a forward operation: message passing (Equation (21)) and readout phases, where M_t—a message function, U_t—vertex update function, h_v^t—hidden state, m_v^{t+1}—message, and T—passing step.

$$\begin{aligned} m_v^{t+1} &= \sum_{w \in N(v)} M_t(h_v^t, h_w^t, e_{vw}) \\ h_v^{t+1} &= U_t(h_v^t, m_v^{t+1}) \end{aligned} \tag{21}$$

  Both message M_t and vertex update functions U_t are implemented using concatenation, Multi-Layer Perceptron (MLP) with ReLu activation, and sum as the aggregation operator (cf. Equations (22) and (23)).

$$M_t = MLP(h_v||h_v||h_{vw});\ w \in N(v) \tag{22}$$

$$U_t = MLP(h_v^T||m_v^{t+1}) \tag{23}$$

- Global Attention Pooling (GAP)—global attention pooling [54] for aggregating node embeddings using attention mechanisms to obtain attention scores and calculate graph embedding Φ. GAP plays the role of MPNN's readout phase.
- Linear Layer (LL)—final linear layer that compresses the graph embedding vector into a singular output, which defines the Q-value of the allocation.

The above approach allows us to aggregate the node/edge information based on the neighbourhood. The role of GAP is to pick the nodes that contribute the most to the graph embedding values, i.e., the traffic concentration nodes. This enables the agent to learn about the relative relationship of the nodes and edges and focus on the nodes/edges that accommodate the highest traffic volume and are at risk of congestion. Combined with the proposed reward and DDQN principles, the agents are encouraged to prioritise the allocations that omit the most loaded edges and nodes, if possible, and load balance across traffic concentration points. Moreover, time is introduced into the model by embedding predictions on node/edge expiry in the network state. A common approach is to combine GNNs with time-series prediction units, e.g., Long Short-Term Memory (LSTM). Hereby, a more generic model is used. While the orbital period of LEO satellites is approximately constant, the periodicity of satellite topology (as visible from the point on the ground) is harder to deduct due to the impact of Earth's rotation and orbital inclination of nodes. This requires careful consideration of the training time window and can potentially lead to progressively lowered model performance as time passes. Moreover, coupling the reasoning with time as is does not take into account random events like interface misalignments, occlusions, or weather conditions that can impact the topology shape and can be predicted by GIMMF-like systems. The 3DQR approach also enables encapsulating the specific

NTN/TN events and time-driven topology changes, leading to a more generic model that can be reused across different 3D systems.

5.4. Complexity Analysis

To evaluate the theoretical complexity of 3DQR E2EP establishment, a popular Shortest Path (SP) implementation is considered that uses priority queues and features time complexity $O(VlogV + E)$. As described in Section 4.2, the E2EP establishment involves a number of operations at the overlay and domain levels, which contribute to the overall 3DQR complexity:

1. Rough E2EP computation by M-SDNC using overlay network graph $G^{\psi}(V^{\psi}, E^{\psi})$;
2. IDP computation by designated SDNCs; using domain network graph $G^{\partial}(V^{\partial}, E^{\partial})$;
3. Verification of E2EP and IDPs feasibility in terms of QoS requirements.

Following the GETPATH procedure definition (cf. Algorithm 1), the complexity is impacted by the calculation of k candidate paths (k shortest paths), the virtual path allocation procedure, and comparison of paths. GETPATH complexity can be expressed as follows:

$$O^{getPath} = k \cdot [O(VlogV + E) + O(E) + O(F)] \tag{24}$$

where $O(E)$ is the worst case for the virtual path allocation procedure (number of hops equal to number of graph edges) and $O(F)$ is the inference complexity of DGA's Q-network. To simplify the above terms, it is approximated that the number of edges is approximately the number of nodes times the average node degree $E \approx deg(V) \cdot V$, leading to the following:

$$\begin{aligned} O^{getPath} &= k \cdot [O(V[logV + deg(V)]) + O(V \cdot deg(V)) + O(F)] \\ &= k \cdot [O(VlogV) + O(V \cdot deg(V) + O(V \cdot deg(V)) + O(F)] \\ &\approx k \cdot [O(VlogV) + O(V \cdot deg(V)) + O(F)] \\ &\approx k \cdot [O(V[logV + deg(V)] + O(F)] \\ &\approx max(O(kV[logV + deg(V)], O(kF)) \end{aligned} \tag{25}$$

The E2EP calculation complexity $O^{getPath}_{total}$ is affected by computing the rough E2EP and IDPs:

$$\begin{aligned} O^{getPath}_{total} = O^{getPath}_{E2EP} + O^{getPath}_{IDP} &= max(O(kV^{\psi}[logV^{\psi} + deg(V^{\psi})], O(kF)) \\ &+ max(O(kV^{\partial}[logV^{\partial} + deg(V^{\partial})], O(kF)) \end{aligned} \tag{26}$$

As described in Section 5.3, the applied DGA model exploits GNNs to aggregate the node/edge information based on the neighbourhood. Hence, the linear increase in network size will not require linear scaling of the model size to maintain performance ($O(F) \approx O(1)$). This allows for omitting the $O(kF)$ term and simplifying Equation (26) to the following:

$$O^{getPath}_{total} = O(kV^{\psi}[logV^{\psi} + deg(V^{\psi})]) + O(kV^{\partial}[logV^{\partial} + deg(V^{\partial})]) \tag{27}$$

Finally, each path undergoes verification regarding QoS constraints on both the local and domain levels, which requires iterating over path segments, giving the worst-case complexity of $O(E)$. Nonetheless, this term can be omitted as the overall complexity of path computation (cf. Equation (27)) will always be dominant, and the path length will usually be much lower than the number of edges. Finally, the overall complexity depends on the number of candidate paths and network graph properties: node degree, number of nodes, topology, and domain split. Figure 5 shows the 3DQR complexity for a different number of domains $|\mathfrak{d}|$. For complexity calculation, the percentage of BNs $\psi_r = 0.2$ (i.e., $V^{\psi} = V \cdot \psi_r$) and an equal split of nodes across domains (i.e., $|V^{\mathfrak{d}}| = \frac{|V|}{|\mathfrak{d}|}$) are assumed. For reference, 3DQR complexity is compared to centralised SP routing and Hierarchical Shortest Path (H-SP) routing (i.e., using SP routing at both overlay and domain levels).

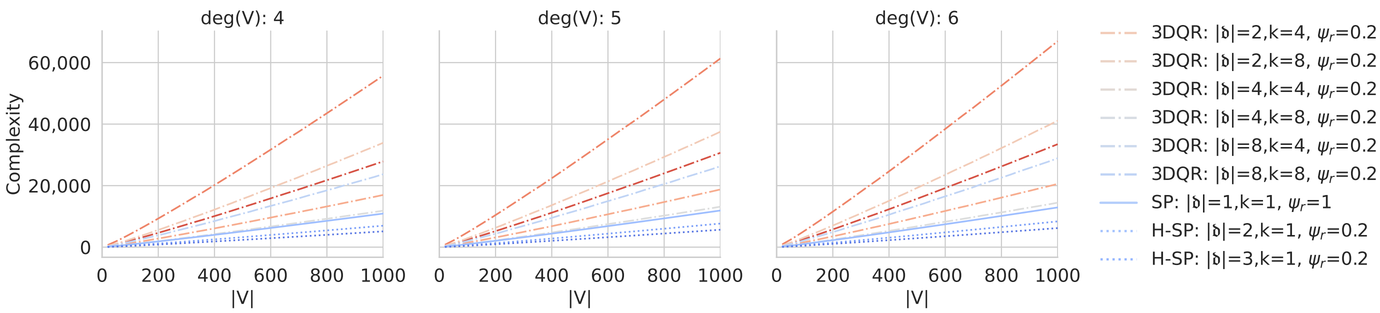


Figure 5. Complexity comparison of 3DQR, H-SP, and SP routing for different average node degrees.

E2EP calculation using 3DQR, in general, features much higher complexity than the other baseline routing schemes. For the increasing number of domains, however, as the operations are distributed across multiple SDNCs, the overall complexity rapidly diminishes. For a low number of computed candidate paths ($k = 4$) and the relatively small number of domains considering real-life networks (i.e., $|\mathfrak{d}| = 8$), 3DQR has almost the same complexity as single-controller SP. Whilst H-SP complexity always surpasses 3DQR, it does so at the cost of worse performance in terms of load distribution and flow acceptance rate as described in Section 6.1.

6. Evaluation

3DQR in the most common case comprising LEO NTN nodes (a subset of Starlink satellites) and synthetic TN mesh-like topologies are evaluated. As LEO satellites constitute the extreme mobility case, the conclusions can be extended over 3D networks due to the typically slow mobility of aerial stratum nodes. The evaluation was conducted using the proprietary event-driven simulator implemented in Python 3.10 and based on SimPy 4.1.1 [55] (hierarchical multi-domain SDN), PyTorch 2.5.0, and PyTorch Geometric 2.6.1 [56] libraries (DRL-based routing). To implement GIMMF, the Skyfield 1.48 library [57] and publicly available Two-Line Element (TLE) files (satellite orbits' descriptors) [58] were used. The 3DQR tests covered three different scopes (cf. Table 1):

- Performance—showing gains of 3DQR compared to three baseline methods: (i) the most-common SP routing using link delay as the weight metric at both overlay and domain levels, further referred to as H-SP; (ii) a combination of SP for overlay routing and NTNs and classic DDQN for TNs (DNN-based architecture), and 3DQR-Uncoordinated (3DQR-U)—domain DGAs and SP-routing at the overlay level (cf. Section 6.1);
- Transfer capabilities—comprising performance tests of DGAs trained in one topology and operating in previously unseen topologies with different topological properties (cf. Section 6.2);
- Aggregation impact—verifying the impact of interval of topology aggregation by GIMMF on 3DQR performance under low traffic load (cf. Section 6.3).

3DQR was tested in over 30 different topologies comprising both sparse and dense networks (T1–T32) with TN nodes located within western and central Europe. The training and performance evaluation (s0-x, s1-x) were conducted in reference topologies comprised of two TN graphs (32 nodes—50% relays, 25% gateways, and 25% gNodeBs) with average node degrees equal to four (T1) and seven (T2); and (ii) two NTN graphs—subsets of

Starlink satellite constellations containing 32 nodes (T1) and 24 nodes (T2), both with an average node degree equal to four (fixed due to limited number of optical interfaces onboard LEOs). To verify transfer capabilities, 20 previously unseen topologies were used, each comprising 16–80 TN and NTN nodes with average node degrees of 3–6. The TN-NTN connections are calculated dynamically based on LEOs proximity to the TN gateway and the visibility criterion (elevation $\geq 30°$). Domain-level links and FLs have a capacity of 100 and 200 kilounits, respectively, $\delta = 10^{-8}$, and delay is calculated based on the spatial distance between nodes. For UE-LEO links, the worst-case delay of 13 ms (the largest value at 600 km orbit as indicated by the 3GPP [48]) and PER of 10^{-7} were adopted. For UE-TN, the Radio Access Network (RAN) delay of 2 ms and $\delta = 10^{-8}$ were assumed.

Table 1. Conducted test scenarios. Transfer capabilities are evaluated using models trained in s0-c and s1-c tests.

Test ID	Topology	Algorithm (TN, NTN, TN-NTN)	Scope
s0-a	T1	H-SP	Performance
s0-b		DDQN-SP-SP	
s0-c		3DQR	
s0-d		3DQR-U	
s1-a	T2	H-SP	
s1-b		DDQN-SP-SP	
s1-c		3DQR	
s1-d		3DQR-U	
s2-a	T1-T23	H-SP	Transfer
s2-b		3DQR (s0-c)	
s2-c		3DQR (s1-c)	
s3	T1	3DQR	Topology aggregation interval 1–40 s

Finally, the adopted QoS classes of arriving flows correspond to expected NTN-based services (cf. Table 2). The arriving flows have a duration of 0.5 to 20 min (generated randomly with uniform distribution) and an equal rate per-QoS class (25% each). In each test, an equal number of flows per domain was generated, i.e., 1/3 inter-domain flows and 2/3 intra-domain flows (equal portions for TN and NTN segments).

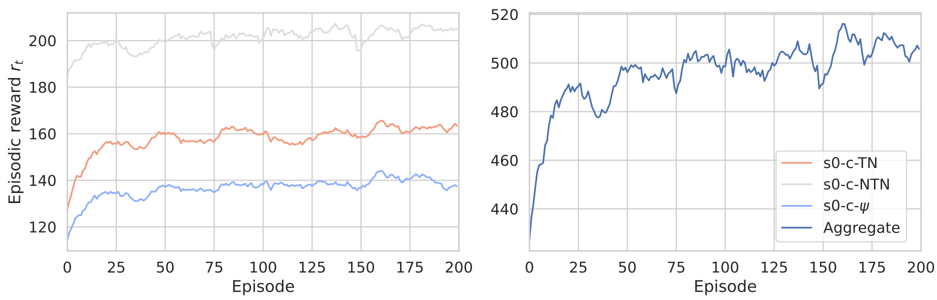
To set up the DRL models, tests were conducted to fine-tune the network architectures and hyperparameters (e.g., k, τ, γ) to maximise the models' performance. As a result, the following configuration was derived for DQRs/M-QR and DGAs: $k = 8$, $\gamma = 0.85$, $\tau = 10^{-2}$, $\epsilon_{min} = 0.01$, $\epsilon_{decay} = 0.995$, $N = 16$, $|t| = 4$, and a learning rate of 10^{-4}. The DGA Q-networks comprise six passes in the MPNN, GAP with MLP (64 hidden neurons and ReLu activation) as a gate function computing attention scores, and one-neuron LL for the compression. The DDQN architecture (s0-b, s1-b) comprises four LLs: 64, 32, 16, and 1 output neurons. GIMMF aggregates the topology for a 5 s window and provides the topological prediction for another 5 s (10 s into the future). In the 3DQR case, the input size equals the number of node and edge features, while in the DDQN case, the input size equals the number of node and edge features multiplied by the number of graph links.

Table 2. QoS profiles used for evaluation (based on [48]).

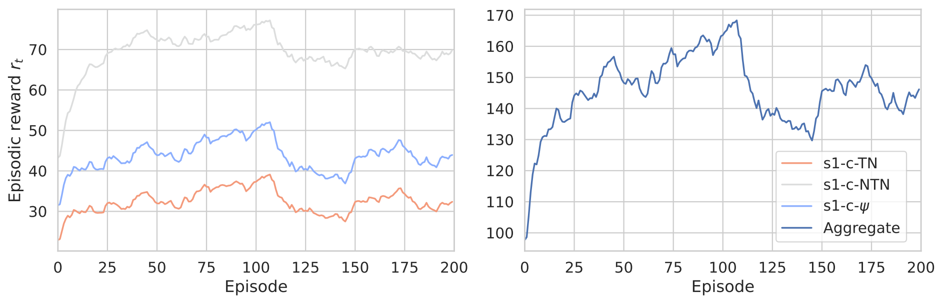
5QI	PDB [ms]	PER	GFBR	MFBR	QF	Example Service
1	100	10^{-2}	75	150	0.3	Conversational voice
2	150	10^{-3}	2000	5000	0.5	Conversational video
4	300	10^{-6}	1000	2000	0.8	Non-conversational video (buffered streaming)
75	50	10^{-2}	500	1000	0.9	A2X messages, aircraft telemetry

6.1. Performance

To evaluate the performance, the impact of the algorithms on network availability and load distribution is assessed. To this end, two metrics are considered: flow rejection rate (fraction of rejected flow allocation requests) and the standard deviation of link utilisation $std(u_{ij})$ expressing the overall imbalance of traffic across all links in a time step. For each test case, the agents were trained in low-traffic environments for 200 episodes lasting 0.5 h each (500 flow requests per episode—$N_{flows} = 500$) and the models were chosen that accumulated the highest aggregate episodic reward. The respective training curves for the 3DQR model are shown in Figure 6.



(**a**) Network topology T1 (s0-c scenario).



(**b**) Network topology T2 (s1-c scenario).

Figure 6. Episodic reward obtained by 3DQR model in low-traffic environment: for domain agents TN, NTN and overlay ψ (**left**); total agents' reward (**right**).

For both scenarios, the model tends to reach satisfactory reward values after about 100 training episodes. For the s1-c scenario, sudden performance degradation can be observed that might have been caused by over-training (due to the smaller state space in the NTN domain). Nonetheless, the model is able to recover after ≈40 episodes to a decent performance. The disproportions across domains and scenarios are caused by different network topologies, the number of flow rerouting operations, and rerouted flow types (inter-/intra-domain, QoS class).

Based on the aggregate reward, the models that performed the best in their respective environments were selected. For further evaluation, the models were tested against the increasing network load, up to $N_{flows} = 3000$, which corresponds to a highly congested network. The results for the best models are shown in Figure 7. First, 3DQR yields a substantial decrease in flow rejection rate compared to H-SP routing for a medium network load and higher ($N_{flows} \geq 1000$). Starting with ≈2% (s0-c, $N_{flows} = 1000$), the reduction reaches up to ≈13.5% (s0-c, $N_{flows} = 3000$). This is due to the ability of consecutive intra-domain and inter-domain optimisation, which allows for offloading feeder links and better exploiting both ISLs and TN infrastructure. This allows 3DQR to allocate the flows evenly across the available links and minimise the risk of congestion. A performance drop can be observed for lower network loads. This is caused by the number of candidate paths each router processed during the flow allocation. Increasing the number of candidate paths increases the potential length of individual network paths (both domain and overlay level), at the same time impacting the overall delay and packet error rate (leading to rejections caused by exceeding the maximum thresholds, i.e., PDB or PER). Nonetheless, the observed degradation is rather small (lower than 2%) and can be potentially mitigated by reducing the number of evaluated candidate paths. Figure 7 also outlines the performance of a combined DNN-based DDQN with H-SP routing and 3DQR-U, where both fail to provide satisfying results. Using the foremost increases the rejection rate up to ≈10%, which indicates overall poor reasoning capabilities and difficulties in converging in the changing network conditions. Regarding 3DQR-U, it yields comparable results to H-SP routing, but at the same time, it improves the load distribution in the network (see further). Nonetheless, 3DQR provides much better results with a minimal increase in complexity, thus emphasising the importance of overlay level TE in the concept. Overall, it must be emphasised that 3DQR provides stable performance regardless of the network load, thus addressing one of the critical issues of DRL-based approaches, i.e., poor performance and unpredictable behaviour in the previously unseen states. 3DQR via mechanisms of MPNN and GAP (cf. Section 5.3) can overcome this problem by first extracting the key features out of the network graph and then appropriately scaling the node features according to their importance, i.e., the risk of causing congestion in the network.

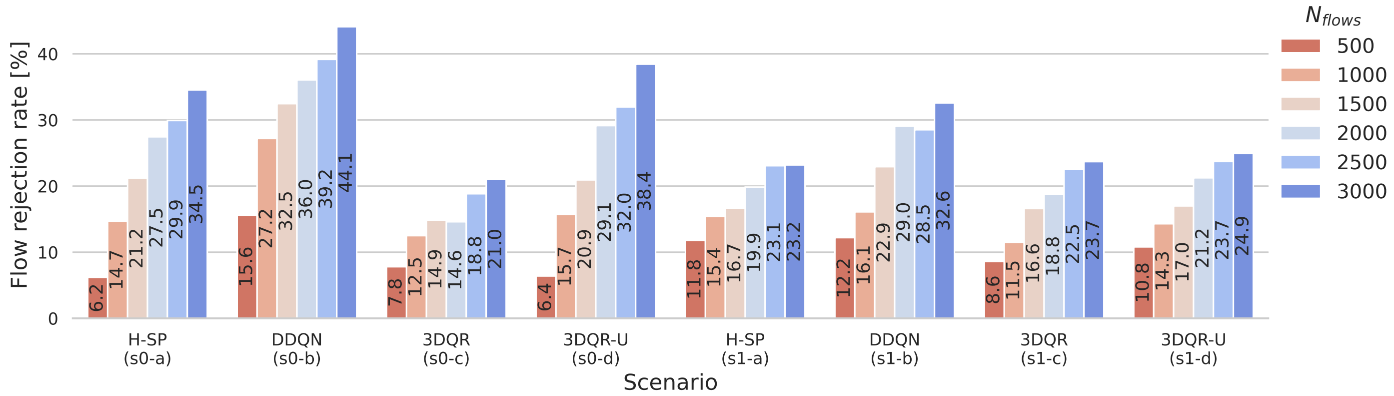


Figure 7. Rejected flows per test setups s0–s1.

The second optimisation target for 3DQR is to prioritise the most advantageous QoS classes (i.e., with the highest QF, cf. Table 2). Figures 8 and 9 show the QoS class rejection rate for each algorithm and traffic load and gains compared to the baseline H-SP routing. First, 3DQR enables prioritisation of the most important 5QI class 75, reducing the drop rate in almost every case up to 4.5% in the congested scenario; however, this occurs at the cost of class 4 (total increase up to 4.5%). QoS class 4, however, consumes the most bandwidth across the classes—it is prone to rejection due to occasional congestions (e.g., due to topological reasons, as it can be observed for s0-c and $N_{flows} = 2000$). Moreover, the drop rate for the remaining classes is significantly lower and increases with the number of flow requests.

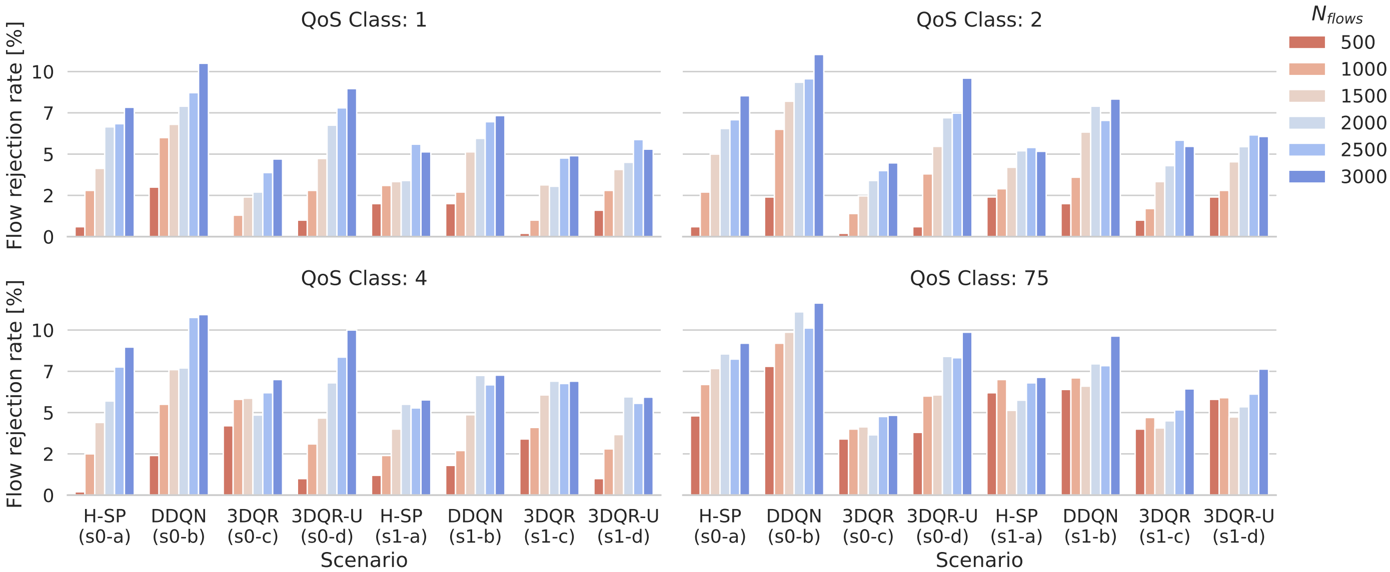


Figure 8. Rejected flows rate for tests per QoS class.

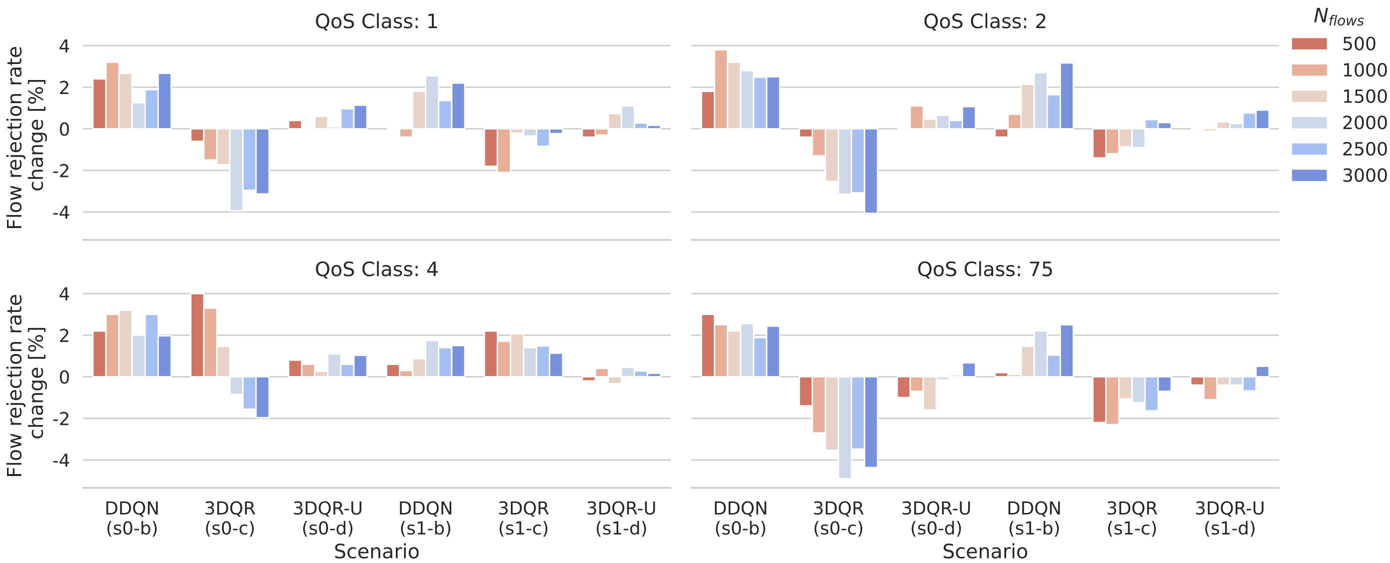


Figure 9. Change of flow rejection rate per QoS class compared to baseline H-SP routing.

The second major goal of 3DQR is to load balance the traffic to improve resource utilisation, especially ISLs and FLs. Figure 10 shows the distributions of $std(u_{ij})$ obtained from the environment during a single episode (aggregate of all segments).

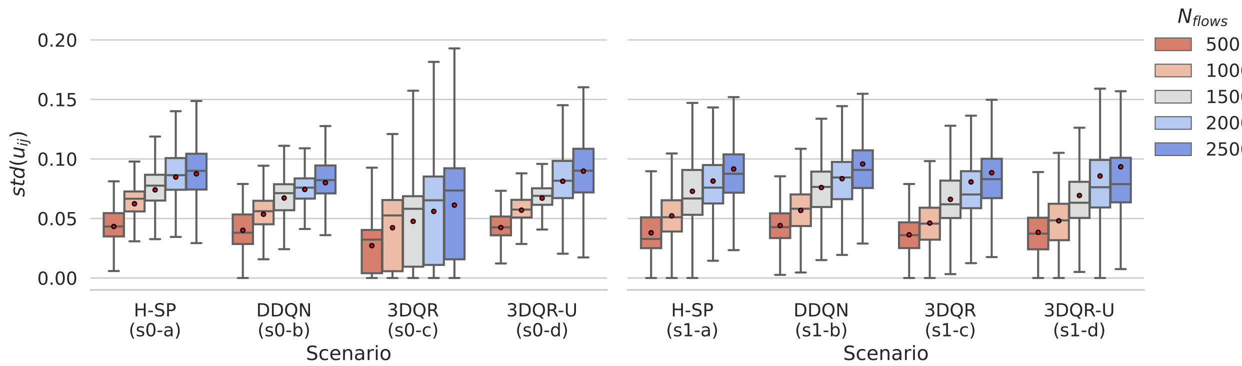


Figure 10. Standard deviation of utilisation u_{ij} of links per domain (mean marked with red dot) for different loads.

First, 3DQR enables a decrease in the mean $std(u_{ij})$, which corresponds to better flow allocation decisions during the episodes (cf. Table 3). For the sparse network, the 3DQR enables up to 52% better distribution of traffic compared to H-SP routing (s0-c). It must be noted that for dense network deployment (s1-c), the gains are consistently lower, but the impact is, respectively, lower. This is caused by the much larger average degree of the network, which leads to evaluating by M-QR multiple shortest paths with a large portion of common links. A potential solution would be to search for the shortest node-disjoint paths to minimise this phenomenon and improve the performance. 3DQR-U, on the other hand, provides consistent gains oscillating around 4–6% for sparse comparable results to H-SP routing for dense networks. A similar trend can be observed for the classical DNN-based DDQN, which indicated the issues of the model to converge in a vast state space. Finally, as shown in Figure 10, 3DQR features lower values of 25th and 75th percentiles for both sparse and dense network scenarios. In the optimal case, the Inter-Quartile Range of $std(u_{ij})$ and mean value equal 0. While this is unachievable in real-life networks, 3DQR features $std(u_{ij})$ promising values (s0-c).

Table 3. The mean (μ) of $std(u_{ij})$ in the best episode (aggregate over domains) and change Δ compared to the baseline scenario (the lower, the better).

N_{flows}	Test	s0-a	s0-b	s0-c	s0-d	s1-a	s1-b	s1-c	s1-d
1000	μ	0.067	0.06	0.044	0.063	0.062	0.065	0.059	0.062
	Δ[%]	-	−11.7	−52.3	−6.3	-	4.6	−5.1	0.0
2000	μ	0.072	0.065	0.048	0.069	0.069	0.073	0.067	0.07
	Δ[%]	-	−10.8	−50.0	−4.3	-	5.5	−3.0	1.4
3000	μ	0.075	0.067	0.051	0.072	0.073	0.077	0.07	0.073
	Δ[%]	-	−11.9	−47.1	−4.2	-	5.2	−4.3	0.0

6.2. Transfer

Here, an attempt to verify the ability of 3DQR to generalise and its reasoning capabilities during the operation in previously unseen topologies is performed. To this end, selected were the best-performance models trained in the T1 and T2 topologies (cf. Section 6.1) and the agents were transferred to over 30 different topologies comprising 16–80 NTN nodes, 32–80 TN nodes, and an average node degree ranging from three to six (sparse to dense network). Also, T1 and T2 topologies were included in the tests to check how well the models native to the environment behave, compared to the transferred ones. Each test consisted of 1500 flow allocation requests with the same configuration described above. The performance comparison is shown in Figure 11.

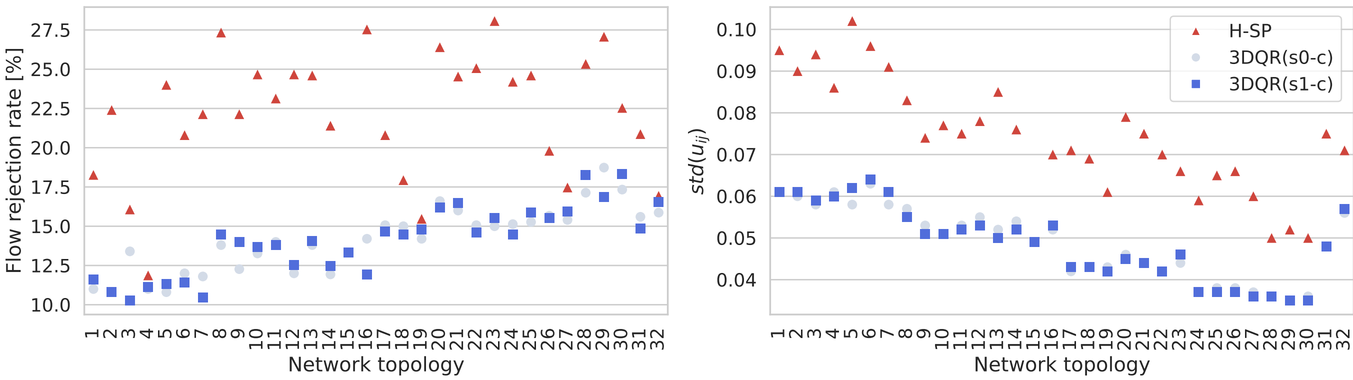


Figure 11. Performance comparison of H-SP-routing and 3DQR model in terms of flow rejection rate (**left**) and load distribution (**right**). Codes 31 and 22 refer to topologies T1 and T2.

First, it can be observed that both of the trained models yield outstanding performance in the vast majority of the considered scenarios, decreasing the flow rejection rate (up to 14%) and improving the load distribution (up to ≈50% reduction in $std(u_{ij})$). Only in four cases was the rejection rate comparable to H-SP routing case, which regarded tests in highly sparse topologies (i.e., node degree equal to three and a small number of TN/NTN nodes) that effectively limited the agents' ability for routing improvements (despite the load distribution gains retained across setups, the congestion occurred). Finally, a very similar performance can be observed for s1-c and s0-c models operating in the T1 and T2 topologies, which further emphasises 3DQR generalisation capabilities. The consistent performance across different setups is a result of using an attention mechanism within the agent—the contribution of edge and node utilisation to the candidate paths Q-values are relevantly scaled, which allows the agents to quickly identify the situations with a risk of congestion to avoid allocations to the expiring links.

6.3. Topology Aggregation Interval Impact

The impact of adopting different aggregation time windows by GIMMF for NTN topology provisioning and predictions is shown in Figure 12. The results show both the rejected flow requests due to no viable QoS paths or no connection between the node pair, e.g., due to a disjoint graph. It can be observed that aggregation interval up to 5 s does not substantially impact rejected and rerouted flows. Increasing the duration of the aggregation interval increases the rejection rate up to 30% for 30 s and longer time frames. This phenomenon occurs due to the dynamicity of LEO topology and the progressively decreasing number of available links in the topology used for routing. An increasing aggregation interval decreases the number of viable links for SDNCs to route the traffic flows, ultimately leading to congestions and a higher probability of unavailable QoS-meeting paths or unreachability due to disjoint network graphs, altogether resulting in higher rejection rates. Moreover, adopting long aggregation intervals limits TE possibilities, which further aggravates the performance drop. Therefore, it is advisable to aggregate topology in rather short time windows to mitigate performance drops and, at the same time, conserve SDNCs computational resources that will be spent on frequent rerouting of flows.

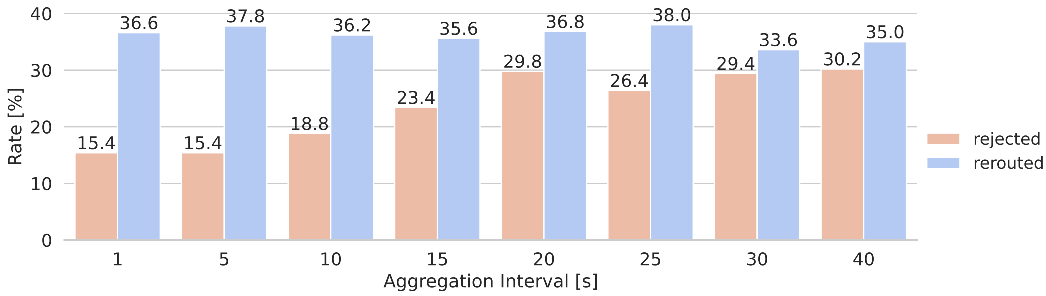


Figure 12. Impact of aggregation interval on flow rerouting and rejection rate.

7. Considerations and Future Work

The 3DQR algorithm enables improving routing decisions, utilising mobile network's resources, and decreasing rejection rates for each class. The practical implementation and deployment of the algorithm, however, pose a set of challenges that need to be considered while deploying the concept in carrier-grade networks:

- **SDN CP distribution and operations granularity—**while hierarchical multi-controller architectures improve the SDN CP scalability, the information exchange needed to establish the E2E paths increases with the degree of distribution. Moreover, it impacts the complexity of an M-SDNC as it needs to synchronise the states and operations of several spatially distant components. Hence, an appropriate deployment strategy and granularity of the distribution need to be adopted, including SDNCs and M-SDNC placement, to avoid large coordination overheads [59].
- **Resource utilisation and flow fairness—**the allocations with fixed QoS guarantees can lead to resource underspending if the allocated resources differ from the ones actually consumed by flows. To this end, it is vital to properly classify the incoming traffic to minimise this effect and employ advanced monitoring mechanisms to obtain actual resource consumption. Moreover, in the case of excessive allocation by the SDNCs, the DP components would require queue-level mechanisms to enforce flow fairness.
- **DRL Performance—**the MADRL setup enables decreasing the state space and variability, which allows the agents to converge faster. In certain cases, however, e.g., in sparse topologies, the observable gain from deploying a DRL optimisation agent might be minor due to limited action space. Hence, the 3DQR deployment should also consider the complexity of individual network segments and DGAs deployment costs.
- **SLA Violations—**due to dynamic conditions, the change of link parameters (e.g., partial ISL occlusion by debris) may result in QoS parameters and SLA violation. Here, allocation itself was focused. However, it is essential to develop monitoring and alerting extensions that allow the tracking of the network status and performing flow rerouting in case of SLA violation risks or mobility events.

8. Summary and Conclusions

This paper presented a novel routing and path allocation framework for future mobile 3D networks called 3DQR. The concept uses hierarchical multi-controller SDN to address key issues of SDN, which involve scalability and interoperability by using operations and metrics abstractions. The framework exploits a MADRL setup for routing based on GNNs and operates on domain and E2E levels, targeting the minimisation of flow rejection and flow rerouting rates and the improvement of load distribution. To improve transferability across different topologies and network configurations, the agent performs the reasoning solely based on the input graph features (node- and link-related SDN DP metrics) and topology prediction data obtained from GIMMF. The conducted evaluation over real and synthetic topologies has proved substantial gains compared to the most common shortest path routing methods regarding traffic distribution (improvement up to 52%) and rejection rate (up to 13.5% smaller) in medium-load to congested network conditions. Finally, considerations regarding deploying the 3DQR concept, possible issues, and perspectives for future improvements have been outlined.

Author Contributions: Conceptualisation, R.K. and R.T.; methodology, R.K.; software, R.K. and R.T.; validation, R.K.; formal analysis, R.K.; investigation, R.K.; data curation, R.K.; writing—original draft preparation, R.K., R.T., L.T. and S.K.; writing—review and editing, R.K., L.T. and S.K.; visualisation, R.K. All authors have read and agreed to the published version of the manuscript.

Funding: ETHER project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101096526. Views and opinions expressed are, however, those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: Authors R.K., R.T., and L.T. were employed by the company Orange Polska S.A. All authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

3DQR	3D QoS-aware Routing
3DQR-U	3DQR-Uncoordinated
3D	Three Dimension
3GPP	3rd Generation Partnership Project
5GS	5G System
5QI	5G QoS Identifier
6G	6th Generation
A-SDNC	Aerial SDNC
AI	Artificial Intelligence
AP	Application Plane
API	Application Programming Interface
BN	Border Nodes
CP	Control Plane
DGA	DRL-GNN Agent
DL	downlink
DNN	Deep Neural Network
DP	Data Plane
DDQN	Double Deep Q-Network
DQR	Domain QoS-aware Router
DRL	Deep Reinforcement Learning
E2E	End-to-End
E2EP	End-to-End Path
ExtReq	External Requester
FL	Feeder Link
GAP	Global Attention Pooling
GEO	Geostationary Earth Orbit
GFBR	Guaranteed Flow Bit Rate
GIMMF	Geographic Information System-based Mobility Management Function
GNN	Graph Neural Network
HAPS	High Altitude Platform System

H-SP	Hierarchical Shortest Path
IDP	Intra-Domain Path
ILP	Integer Linear Programming
IP	Internet Protocol
ISL	Inter-Satellite Link
LEO	Low Earth Orbit
LL	Linear Layer
LSTM	Long Short-Term Memory
M-QR	Master QR
M-SDNC	Main SDNC
MADRL	Multi-Agent Deep Reinforcement Learning
MANO	Management and Orchestration
MDP	Markov Decision Process
MEO	Medium Earth Orbit
MFBR	Maximum Flow Bit Rate
ML	Machine Learning
MLP	Multi-Layer Perceptron
MPLS-TE	MultiProtocol Label Switching-Traffic Engineering
MPNN	Message Passing Neural Network
NBI	NorthBound Interface
NTN	Non-Terrestrial Network
OF	OpenFlow
OSPF	Open Shortest Path First
PDB	Packet Delay Budget
PER	Packet Error Rate
QF	QoS Factor
QoS	Quality of Service
RAN	Radio Access Network
RL	Reinforcement Learning
RC	Rerouting Cost
S-SDNC	Satellite SDNC
SCN	Service-Customised Network
SAGIN	Space-Air-Ground Integrated Network
SDN	Software Defined Networking
SDNC	SDN Controller
SLA	Service Level Agreement
SotA	State of the Art
SP	Shortest Path
SR	Source Routing
T-SDNC	Terrestrial SDNC
TCP	Transport Control Protocol
TE	Traffic Engineering
TED	Traffic Engineering Database
TLE	Two-Line Element
TN	Terrestrial Network
UAV	Unmanned Aerial Vehicle
UE	User Equipment
UL	uplink
UP	User Plane

References

1. ITU-R. *Future Technology Trends of Terrestrial International Mobile Telecommunications Systems Towards 2030 and Beyond*; Report M.2516-0; International Telecommunication Union—Radiocommunication Sector: Geneva, Switzerland, 2022.

2. 3GPP. Study on Using Satellite Access in 5G, ver. 16.0.0; Technical Report TR 22.822; 3rd Generation Partnership Project. 2018. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3372 (accessed on 2 March 2025).
3. Guidotti, A.; Vanelli-Coralli, A.; Schena, V.; Chuberre, N.; El Jaafari, M.; Puttonen, J.; Cioni, S. The Path to 5G-Advanced and 6G Non-Terrestrial Network Systems. In Proceedings of the 2022 11th Advanced Satellite Multimedia Systems Conference and the 17th Signal Processing for Space Communications Workshop ASMS/SPSC), Graz, Austria, 6–8 September 2022; pp. 1–8. [CrossRef]
4. Tomaszewski, L.; Kołakowski, R.; Mesodiakaki, A.; Ntontin, K.; Antonopoulos, A.; Pappas, N.; Fiore, M.; Mosahebfard, M.; Watts, S.; Harris, P.; et al. ETHER: Energy- and Cost-Efficient Framework for Seamless Connectivity over the Integrated Terrestrial and Non-terrestrial 6G Networks. In Proceedings of the Artificial Intelligence Applications and Innovations. AIAI 2023 IFIP WG 12.5 International Workshops, León, Spain, 14–17 June 2023; Maglogiannis, I., Iliadis, L., Papaleonidas, A., Chochliouros, I., Eds.; pp. 32–44. [CrossRef]
5. King, D.; Shortt, K. *Time Variant Challenges for Non-Terrestrial Networks*; Internet Draft draft-king-tvr-ntn-challanges-00; Internet Engineering Task Force: Wilmington, DE, USA, 2023.
6. Ali, I.; Al-Dhahir, N.; Hershey, J. Predicting the Visibility of LEO Satellites. *IEEE Trans. Aerosp. Electron. Syst.* **1999**, *35*, 1183–1190. [CrossRef]
7. Altamirano, J.C.; Slimane, M.A.; Hassan, H.; Drira, K. QoS-aware Network Self-management Architecture Based on DRL and SDN for Remote Areas. In Proceedings of the 2022 IEEE 11th IFIP International Conference on Performance Evaluation and Modeling in Wireless and Wired Networks (PEMWN), Rome, Italy, 8–10 November 2022; pp. 1–6. [CrossRef]
8. Guo, Y.; Lin, B.; Tang, Q.; Ma, Y.; Luo, H.; Tian, H.; Chen, K. Distributed Traffic Engineering in Hybrid Software Defined Networks: A Multi-Agent Reinforcement Learning Framework. *IEEE Trans. Netw. Serv. Manag.* **2024**, *21*, 6759–6769. [CrossRef]
9. Chen, X.; Ji, Z.; Wu, S.; Jia, H.; Xiao, A.; Jiang, C. A Distributed Routing Algorithm for LEO Satellite Networks: A Multi-Agent Transformer-MIX Learning Approach. *IEEE Internet Things J.* **2025**, *Early Access*. [CrossRef]
10. Liao, H.; Zhang, X.; Zhou, J.; Li, X. Real-Time Routing Design for LEO Satellite Networks: An Enhanced Multi-Agent DRL Approach. In Proceedings of the 2024 IEEE/CIC International Conference on Communications in China (ICCC Workshops), Hangzhou, China, 7–9 August 2024; pp. 547–552. [CrossRef]
11. Liu, X.; Chen, A.; Zheng, K.; Chi, K.; Yang, B.; Taleb, T. Distributed Computation Offloading for Energy Provision Minimization in WP-MEC Networks with Multiple HAPs. *IEEE Trans. Mob. Comput.* **2024**, *Early Access*. [CrossRef]
12. Ran, Y.; Ding, Y.; Chen, S.; Lei, J.; Luo, J. Fully-Distributed Dynamic Packet Routing for LEO Satellite Networks: A GNN-Enhanced Multi-Agent Reinforcement Learning Approach. *IEEE Trans. Veh. Technol.* **2024**, *Early Access*. [CrossRef]
13. Ammar, S.; Pong Lau, C.; Shihada, B. An In-Depth Survey on Virtualization Technologies in 6G Integrated Terrestrial and Non-Terrestrial Networks. *IEEE Open J. Commun. Soc.* **2024**, *5*, 3690–3734. [CrossRef]
14. Bannour, F.; Souihi, S.; Mellouk, A. Distributed SDN Control: Survey, Taxonomy, and Challenges. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 333–354. [CrossRef]
15. Guo, J.; Yang, L.; Rincon, D.; Sallent, S.; Fan, C.; Chen, Q.; Li, X. SDN Controller Placement in LEO Satellite Networks Based on Dynamic Topology. In Proceedings of the 2021 IEEE/CIC International Conference on Communications in China (ICCC), Xiamen, China, 28–30 July 2021; pp. 1083–1088. [CrossRef]
16. Kołakowski, R.; Kukliński, S.; Tomaszewski, L. Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Networks. In Proceedings of the 2024 IFIP Networking Conference (IFIP Networking), Thessaloniki, Greece, 3–6 June 2024; pp. 607–612. [CrossRef]
17. Geraci, G.; Lopez-Perez, D.; Benzaghta, M.; Chatzinotas, S. Integrating Terrestrial and Non-terrestrial Networks: 3D Opportunities and Challenges. *IEEE Commun. Mag.* **2022**, *61*, 42–48. [CrossRef]
18. Han, L.; Retana, A.; Westphal, C.; Li, R. Large Scale LEO Satellite Networks for the Future Internet: Challenges and Solutions to Addressing and Routing. *Comput. Netw. Commun.* **2022**, *1*, 30–57. [CrossRef]
19. Du, P.; Nazari, S.; Mena, J.; Fan, R.; Gerla, M.; Gupta, R. Multipath TCP in SDN-enabled LEO Satellite Networks. In Proceedings of the MILCOM 2016–2016 IEEE Military Communications Conference, Baltimore, MD, USA, 1–3 November 2016; pp. 354–359. [CrossRef]
20. Monzon Baeza, V.; Rigazzi, G.; Aguilar, S.; Ferrus, R.; Ferrer, J.; Mhatre, S.; Guadalupi, M. IoT-NTN Communications via Store-and-Forward Core Network in Multi-LEO-satellite Deployments. In Proceedings of the 2024 IEEE 35th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Valencia, Spain, 2–5 September 2024; pp. 1–6. [CrossRef]
21. 3GPP. Study on Satellite Access—Phase 3, ver. 19.2.0. Technical Report TR 22.865. 3rd Generation Partnership Project. 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4089 (accessed on 2 March 2025).
22. Orange. *Mobile Network Technology Evolutions Beyond 2030*; White Paper; Orange: Paris, France, 2024.

23. Yang, Z.; Li, H.; Wu, Q.; Wu, J. Topology Discovery Sub-Layer for Integrated Terrestrial-Satellite Network Routing Schemes. *China Commun.* **2018**, *15*, 42–57. [CrossRef]
24. Cao, X.; Li, Y.; Xiong, X.; Wang, J. Dynamic Routings in Satellite Networks: An Overview. *Sensors* **2022**, *22*, 4552. [CrossRef] [PubMed]
25. Korikawa, T.; Takasaki, C.; Hattori, K.; Oowada, H. Time-Topology Routing in 3D Networks. In Proceedings of the 2023 International Conference on Computing, Networking and Communications (ICNC), Honolulu, HI, USA, 20–22 February 2023; pp. 348–352. [CrossRef]
26. Kumar, P.; Bhushan, S.; Halder, D.; Baswade, A.M. fybrrLink: Efficient QoS-aware Routing in SDN Enabled Next-Gen Satellite Networks. *arXiv* **2021**, arXiv:2106.07778. [CrossRef]
27. Jiang, Y.; Wu, S.; Mo, Q. A Compass Time-Space Model-Based Virtual IP Routing Scheme for NTSN Satellite Constellations. *Chin. J. Aeronaut.* **2023**, *36*, 280–288. [CrossRef]
28. Yang, Z.; Liu, H.; Jin, J.; Tian, F. A Cooperative Routing Algorithm for Data Downloading in LEO Satellite Network. In Proceedings of the 2021 IEEE 21st International Conference on Communication Technology (ICCT), Tianjin, China, 13–16 October 2021; pp. 1386–1391. [CrossRef]
29. Liu, Q.; Li, X.; Ji, H.; Zhang, H. Multi-Path Routing Algorithm with Joint Optimization of Load-Balancing for Cluster-Based Leo Satellite Networks. In Proceedings of the 2023 8th IEEE International Conference on Network Intelligence and Digital Content (IC-NIDC), Beijing, China, 3–5 November 2023; pp. 264–268. [CrossRef]
30. Rao, Y.; Wang, R. Multi-Path QoS Routing Using Genetic Algorithm for LEO Satellite Networks. *Chin. J. Electron.* **2011**, *20*, 17–20.
31. Lian, P.; Yan, F.; Luo, H.; Wang, Z.; Zhang, S. Multicast Source Routing Based on Bloomed Link Identifiers for LEO Satellite Network. In Proceedings of the 2022 IEEE International Conference on Satellite Computing (Satellite), Shenzhen, China, 26–27 November 2022; pp. 13–18. [CrossRef]
32. Ventre, P.L.; Tajiki, M.M.; Salsano, S.; Filsfils, C. SDN Architecture and Southbound APIs for IPv6 Segment Routing Enabled Wide Area Networks. *IEEE Trans. Netw. Serv. Manag.* **2018**, *15*, 1378–1392. [CrossRef]
33. Tang, F.; Mao, B.; Kawamoto, Y.; Kato, N. Survey on Machine Learning for Intelligent End-to-End Communication Toward 6G: From Network Access, Routing to Traffic Control and Streaming Adaption. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1578–1598. [CrossRef]
34. Wang, C.; Wang, H.; Wang, W. A Two-Hops State-Aware Routing Strategy Based on Deep Reinforcement Learning for LEO Satellite Networks. *Electronics* **2019**, *8*, 920. [CrossRef]
35. Tsai, K.C.; Fan, L.; Wang, L.C.; Lent, R.; Han, Z. Multi-Commodity Flow Routing for Large-Scale LEO Satellite Networks Using Deep Reinforcement Learning. In Proceedings of the 2022 IEEE Wireless Communications and Networking Conference (WCNC), Austin, TX, USA, 10–13 April 2022; pp. 626–631. [CrossRef]
36. Geyer, F.; Carle, G. Learning and Generating Distributed Routing Protocols Using Graph-Based Deep Learning. In Proceedings of the 2018 Workshop on Big Data Analytics and Machine Learning for Data Communication Networks, Budapest Hungary, 20 August 2018; pp. 40–45. [CrossRef]
37. Rusek, K.; Suarez-Varela, J.; Almasan, P.; Barlet-Ros, P.;Cabellos-Aparicio, A. RouteNet: Leveraging Graph Neural Networks for Network Modeling and Optimization in SDN. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 2260–2270. [CrossRef]
38. Zhuang, Z.; Wang, J.; Qi, Q.; Sun, H.; Liao, J. Toward Greater Intelligence in Route Planning: A Graph-Aware Deep Learning Approach. *IEEE Syst. J.* **2020**, *14*, 1658–1669. [CrossRef]
39. Dudukovich, R.; Hylton, A.; Papachristou, C. A Machine Learning Concept for {DTN} Routing. In Proceedings of the 2017 IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE), Montreal, QC, Canada, 10–12 October 2017; pp. 110–115. [CrossRef]
40. Zhang, S.; Yin, B.; Zhang, W.; Cheng, Y. Topology Aware Deep Learning for Wireless Network Optimization. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 9791–9805. [CrossRef]
41. Almasan, P.; Suárez-Varela, J.; Rusek, K.; Barlet-Ros, P.; Cabellos-Aparicio, A. Deep Reinforcement Learning Meets Graph Neural Networks: Exploring a Routing Optimization Use Case. *Comput. Commun.* **2022**, *196*, 184–194. [CrossRef]
42. He, C.; Balasubramanian, K.; Ceyani, E.; Yang, C.; Xie, H.; Sun, L.; He, L.; Yang, L.; Yu, P.S.; Rong, Y.; et al. FedGraphNN: A Federated Learning System and Benchmark for Graph Neural Networks. *arXiv* **2021**, arXiv:2104.07145. [CrossRef]
43. Eiza, M.; Raschellà, A. A Hybrid SDN-based Architecture for Secure and QoS Aware Routing in Space-Air-Ground Integrated Networks (SAGINs). In Proceedings of the 2023 IEEE Wireless Communications and Networking Conference (WCNC), Glasgow, UK, 26–29 March 2023. [CrossRef]
44. Wei, L.; Shuai, J.; Liu, Y.; Wang, Y.; Zhang, L. Service Customized Space-Air-Ground Integrated Network for Immersive Media: Architecture, Key Technologies, and Prospects. *China Commun.* **2022**, *19*, 1–13. [CrossRef]
45. Zhang, N.; Zhang, S.; Yang, P.; Alhussein, O.; Zhuang, W.; Shen, X.S. Software Defined Space-Air-Ground Integrated Vehicular Networks: Challenges and Solutions. *IEEE Commun. Mag.* **2017**, *55*, 101–109. [CrossRef]

46. ONF. *OpenFlow Switch Specification, Version 1.5.1 (Protocol Version 0x06)*; Specification ONF TS-025; Open Networking Foundation: Palo Alto, CA, USA, 2015.
47. van Hasselt, H.; Guez, A.; Silver, D. Deep Reinforcement Learning with Double Q-learning. In Proceedings of the 30th AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2015. [CrossRef]
48. 3GPP. System Architecture for the 5G System (5GS), ver. 19.2.1. Technical Standard TS 23.501. 3rd Generation Partnership Project. 2025. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3144 (accessed on 2 March 2025).
49. Phemius, K.; Bouet, M. Monitoring Latency with OpenFlow. In Proceedings of the 9th International Conference on Network and Service Management (CNSM 2013), Zurich, Switzerland, 14–18 October 2013; pp. 122–125. [CrossRef]
50. Wang, J.; Liu, J.; Guo, H.; Mao, B. Deep Reinforcement Learning for Securing Software-Defined Industrial Networks With Distributed Control Plane. *IEEE Trans. Ind. Inform.* **2022**, *18*, 4275–4285. [CrossRef]
51. Tang, M.; Cai, S.; Lau, V.K.N. Online System Identification and Control for Linear Systems with Multiagent Controllers Over Wireless Interference Channels. *IEEE Trans. Autom. Control* **2023**, *68*, 6020–6035. [CrossRef]
52. van Hasselt, H. Double Q-learning. In *Advances in Neural Information Processing Systems 23: Proceedings of the 24th Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 6–9 December 2010*; Lafferty, J.D., Williams, C.K.I., Shawe-Taylor, J., Zemel, R.S., Culotta, A., Eds.; Curran Associates, Inc.: New York, NY, USA, 2010; Volume 3, pp. 2613–2621.
53. Gilmer, J.; Schoenholz, S.S.; Riley, P.F.; Vinyals, O.; Dahl, G.E. Neural Message Passing for Quantum Chemistry. In Proceedings of the 34th International Conference on Machine Learning—Volume 70, ICML'17, Sydney, NSW, Australia, 6–11 June 2017; pp. 1263–1272. Available online: https://dl.acm.org/doi/pdf/10.5555/3305381.3305512 (accessed on 2 March 2025).
54. Lee, J.; Lee, I.; Kang, J. Self-Attention Graph Pooling. *arXiv* **2019**, arXiv:1904.08082. [CrossRef]
55. SimPy Documentation. Available online: https://simpy.readthedocs.io/en/latest/ (accessed on 2 March 2025).
56. PyG Documentation—Pytorch_geometric Documentation. Available online: https://pytorch-geometric.readthedocs.io/en/latest/ (accessed on 2 March 2025).
57. Skyfield—Documentation. Available online: https://rhodesmill.org/skyfield/ (accessed on 2 March 2025).
58. CelesTrak: Current Supplemental GP Element Sets. Available online: https://celestrak.org/NORAD/elements/supplemental/ (accessed on 2 March 2025).
59. Gunther, N.J. A Simple Capacity Model of Massively Parallel Transaction Systems. In Proceedings of the 19th International Computer Measurement Group Conference, San Diego, CA, USA, 6–10 December 1993.

Time-of-Day-Aware Slice Admission Control

Robert Kołakowski*◇, Sławomir Kukliński*◇, Lechosław Tomaszewski*
*Orange Polska, ◇Warsaw University of Technology
Warsaw, Poland

***Abstract*—Network slicing is expected to be a core feature of modern telecommunication networks as it enables the creation of multiple virtual networks sharing a common infrastructure. An efficient Slice Admission Control (SAC) mechanism is critical for the massive deployment of network slices. This paper presents a novel, Deep Reinforcement Learning-based SAC algorithm that considers network users' activity patterns and resulting daily resource demand changes to make admission decisions. Compared to conventional approaches, the proposed solution enables a higher slice admission rate with limited customer Service Level Agreement (SLA) violations due to more efficient usage of available resources, thus facilitating the increase of operator revenue.**

***Index Terms*—network slicing, slice admission control, RL, DRL, DQN, double DQN, traffic pattern, 5G, AI, dynamic resource management**

I. Introduction

The Network Slicing (NS) concept is a fundamental paradigm change in mobile networks arriving with the Standalone variant of 5G System (5GS). In contrast to the hitherto versatile mobile network to support all services, 5GS allows the creation of multiple network slices supporting specific classes of services or use cases. The network slices, despite their isolation, share a common resource pool, typically having finite resources. This leads to the problem of proper resource allocation to slices and Slice Admission Control (SAC). SAC is responsible for accepting or rejecting NSI instantiation requests, taking into account required resources, priority, Slice/Service Type (SST). The efficient SAC strategy is an essential part of resource management for Mobile Network Operators (MNOs) for multiple reasons, e.g., users' Service Level Agreement (SLA) assurance, maximizing profit from accepted Network Slice Instances (NSIs), minimizing SLA violations, network congestion control, ensuring fairness to the slice tenants, etc. [1]. One of the promising approaches to SAC is the exploitation of the Reinforcement Learning (RL) algorithms, which can capture both short- and long-term dependencies between the arriving NSI requests and the profit from the NSIs chosen to be deployed. Multiple studies have shown significant benefits of such an approach over the traditional greedy methods [2], [3]. The State of the Art (SotA) SAC algorithms ignore the fact that the resources consumed by an NSI can be related to human activity, i.e., are dependent on the time of day. This paper proposes a novel Double Deep Q-Network (DDQN)-based SAC algorithm, Time-of-Day-Aware Slice Admission Control (TASAC), that uses the information about the NSIs' requested resource allocation, slice type, as well as current and predicted resource consumption, with aims to maximize the overall utilization of network resources while minimizing SLA violations. The predictions are driven by the users' activity models in the day-night cycles to predict both, the individual slice consumption and the available resource pool more accurately, resulting in robust SAC decisions and more efficient resource utilization. The model is typically applicable for short-lived slices (< 1 day).

II. Related work

The future mobile networks are expected to accommodate many dynamically deployed NSIs with different requirements and activity patterns. Hence, efficient resource management via optimal SAC will be key to increasing infrastructure utilization and MNOs' profits. Recently, research activities have proposed various SAC algorithms, resource allocation schemes, resource usage prediction, and slice modeling [1].

The foundation techniques of SAC involve RL and Deep Reinforcement Learning (DRL) based algorithms. A framework for slice brokering called "RL-NSB" is proposed in [4]. The admission control is formulated as an NP-hard, geometric two-dimensional knapsack problem, and heuristic algorithms are proposed to address regular and irregular arrival of NSI requests. Another approach capitalizing on DRL to solve both SAC and Resource Allocation (RA) problems using Q-Learning (QL) and Deep Q-Network (DQN) algorithms is defined [2]. The DQN method enabled over 3.6% greater profit than QL and baseline methods (7.9% for the Node Ranking algorithm, 11.7% for an Always Admit Request method). The benefit of using RL-based SAC has also been proved [3], where QL, DQN and the online algorithm called Regret Matching (RM) aiming to maximize the revenue from deployed NSIs are introduced. The algorithms provided satisfactory results, but RM brought the highest financial gain, supposedly due to the regret formula allowing faster policy adaptation in case of negative rewards. The use of the Markov Chain Process to model SAC problem can also be found in the literature [5]. An analysis of the system modeled as a Semi-Markov Decision Process and design of DRL method maximizing infrastructure providers' revenue and Quality of Service (QoS) control has been presented [6]. The RL-based SAC strategy targeting 5G flexible Radio Access Network (RAN) and operating in the presence of slices of various MNOs and with different priorities is shown [7]. The proposed policy learns the potential high-profit services (maximized revenue with low degradation) and should be accepted.

The SotA of SAC approaches do not consider the dependencies between Time-of-Day (ToD) of slice operation and the number of consumed resources explicitly, focusing only on the frequency of arriving Slice Admission Requests (SARs) and making fixed resource allocations. The resource demand and time relationship has already been observed and modeled for the previous generations of mobile networks [8]. The stochastic Markov traffic model [9] derived from the traces from the Long Term Evolution (LTE) network scheduling information acquired in a European metropolis is proposed. Another approach introduces the sinusoidal superposition model [10], evaluated using the real data obtained from 185 base stations in a dense urban area in a city in China and proving decent accuracy. Numerical models to estimate aggregate traffic in the 5G network also show the characteristic changes of intensity of human Enhanced Mobile Broadband (eMBB) and Massive Machine Type Communications (mMTC) traffic [11].

The exploitation of the slice-specific ToD activity can lead to a much more efficient and realistic SAC than ToD activity pattern-agnostic approaches. The proposal of a ToD-aware SAC is presented below.

III. Concept Description

From the MNOs' perspective, one of the core benefits of the NS concept is the ability to provide diverse communication services using common infrastructural resources (typically virtualized connectivity, memory, and computation). The key mechanisms to optimize the utilization of MNOs' infrastructure include efficient and accurate NSI resources allocation (including scaling) and the SAC mechanism. As in 3rd Generation Partnership Project (3GPP) networks, the foremost process is automated by the European Telecommunications Standards Institute (ETSI) Management and Orchestration (MANO) orchestrator. Hereby, we focus only on defining the SAC strategy to improve the utilization of MNOs' resources.

The complexity of NS requested by the NSI tenants can be immense and involve various requirements referring to NS operation, such as capacity, spatial and time availability, supporting applications, security features, and more. Our concept focuses only on the orchestration aspect of each SAR, as the above high-level slice-specific features can be translated into low-level physical requirements. In general, the resource consumption of each NSI depends on two main factors:

- resources needed for the deployment of NSI (i.e., its footprint) and its basic operations in an "idle" state;
- resources needed to support the activity of NSI-attached terminals, out of which some can be human-operated.

It can be noted that human-generated network traffic is ToD-dependent (at night, network load is typically lower than during the day, cf. Fig. 1) and can be further split into business- and home-oriented applications. Such periodicity of human behavior is well known and exploited by MNOs designing their networks to handle peak-hours traffic. In contrast to human-operated terminals, the Internet of Things (IoT), Industry 4.0 terminals, and security cameras traffic is almost constant over time and doesn't exhibit the ToD dependency.

The knowledge of ToD resource consumption can be exploited in SAC:

- for long-lived NSIs (duration > 24 h) by predicting ToD peak, it is possible to reduce the probability of congestion during peak hours;
- for short-lived NSIs (duration < 24 h), it is possible to allocate more resources depending on ToD; SAC can be more aggressive on the descending slope of ToD and less aggressive on the ascending one.

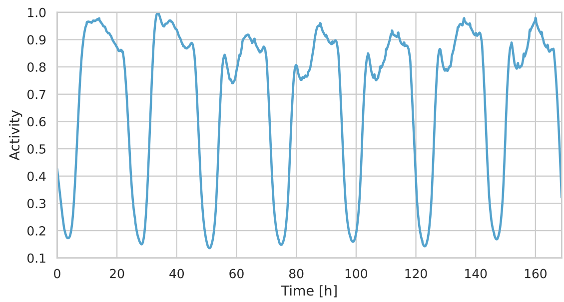


Fig. 1. Normalized traffic intensity (obtained from the production network)

In our analysis, we consider traffic generated by both human- and non-human-operated terminals, referring to them as eMBB and mMTC, respectively. eMBB SST is treated as ToD-dependent (i.e., scaling with time), while mMTC SST is treated as ToD-independent. Each SAR is specified in terms of the above-mentioned SSTs and is accompanied by additional relevant parameters related to the NSI deployment specifics.

A. *TASAC functional blocks*

The functional blocks that compose the TASAC concept are depicted in Fig. 2. NSI tenants send SARs containing the high-level information (duration, deployment time, capacity) about NSI to be deployed in the network. The requests are transferred to the Resource Mapper entity, which maps the generic request into the slice resource requirements and priorities according to the capacity preferences and SST, as in Generic network Slice Template (GST)/Network Slice Template (NEST) approach [12]. The mapped SAR is put into Requests Queue (priority queue) to be analyzed by Slice Admission Engine (SAE), which implements the SAC policy and makes slice admission/rejection decisions based on:

- Current resource consumption – total resource utilization data provided by Resource Consumption Monitor (system monitoring layer);
- System status – information on deployed NSIs, their types, peak resource requirements, remaining resources, etc. It is stored in the Admission Database and updated on system changes (SAR acceptance, NSI termination);
- Prediction of resource consumption – it is assumed that various prediction techniques can be used to deliver accurate predictions depending on the circumstances. In TASAC, this data is supplied by Resource Consumption Predictor, which leverages the information about the ToD activity curves relevant to SSTs for forecasting.

SAE encapsulates a DDQN-based agent responsible for deriving the SAC policy during the system operation. The novelty of TASAC lies in enhancing the well-established algorithm by incorporating the ToD activity changes and its predictions to improve the performance. Note that the TASAC algorithm makes some simplifications. First, we use the resource as a scalar instead of a vector (e.g., computing, storage, and connectivity). Such simplification has no impact on the algorithm behavior and provides better re-usability (if analysis of multiple resource types is required, SAC can be extended by, e.g., ensemble and voting methods). SST is used by the SAC algorithm to estimate the future NSI resource needs fluctuations. In the case of mMTC and IoT NSIs, it is assumed that resource consumption is changing slowly, i.e., it is treated as ToD independent. For other SSTs, the scaling of resources is assumed to be correlated with the ToD curve.

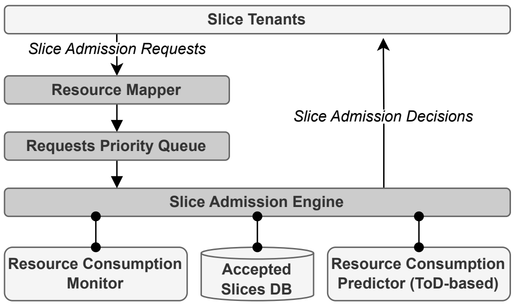


Fig. 2. ToD-based slice admission process – concept overview

The main uncertainty regarding NSI deployment is related to the need of resource scaling, i.e., dynamic allocation of resources to NSI during its runtime. In this analysis, it is modeled as ToD-dependent variable for human-oriented slices and nearly constant for machine-to-machine SSTs, i.e., mMTC or Ultra-Reliable Low-Latency Communication (URLLC). While the ToD curve shape is generally known [9]–[11], its exact values have to be estimated. For the sake of simplicity, it is assumed that the behavior of users is the same during the whole week. However, a simple modification of the approach, i.e., adding separate curves for non-labor days, religious or national holidays, etc., can lead to more accurate results. Exploiting the ToD resource curves allows decision-making fitter to the actual network conditions. Naturally, when SAR comes at the ToD resource consumption rising slope, the probability of its acceptance should be much lower than the acceptance of the same SAR on the descending slope.

IV. Time-of-Day-Aware Slice Admission Control

The goal of the TASAC algorithm is to exploit the knowledge of ToD-dependent consumption by the slices to improve the resources utilization during the day by reusing pre-allocated NSIs resources but not consumed off the peak-hours. To derive admission policy, TASAC uses the DDQN algorithm as the basis (cf. section IV-A) and extends the traditional approach to incorporate the ToD user activity patterns (cf. section IV-B). The resources needed for NSI deployment are typically split into computing, storage, and connectivity. As the consumption of each resource type will scale differently with the activity, in TASAC, we treat the resources in a generic way (measured in units) for flexible adaptation.

A. Double Deep Q-Networks

DDQN [13] is a model-free, off-policy algorithm, based on the DQN [14] approach. The primary concept underlying the DDQN is the QL algorithm [15]. We consider a function $Q^* : State\ x\ Action \rightarrow \mathbb{R}$, which provides the information about the value of the actions in each environment's state. Knowing the exact formula enables constructing the policy to maximize rewards for taken actions $\pi^*(s) : argmax\ Q^*(s, a)$. Due to the environment's complexity, deriving the Q^* function is often problematic. The DDQN solves this issue by using Deep Neural Network (DNN) for its approximation.

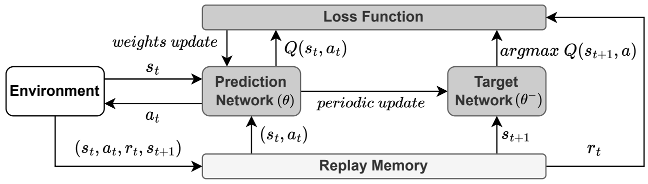


Fig. 3. High level concept of the DDQN approach

The DDQN agent is composed of 4 blocks (cf. Fig. 3), Prediction Network (PN), Target Network (TN), the Loss Calculation block, and the Replay Memory. Both PN and TN implement the same DNN architecture, with weights θ and θ^-, respectively. Every step t, the agent obtains information about the environment state s_t. PN calculates the Q-values corresponding to the separate actions based on the s_t. The action a_t is taken by the agent based on the Q-values (or randomly in the exploration phase) and executed on the environment, leading to a transition to state s_{t+1} and obtaining reward r_t. The information about the transition and reward is stored in the Replay Memory for use during PN training. The update follows the modified Bellman equation (cf. eq. 1) where γ is a decay coefficient to calculate future discounted rewards.

$$Q^*(s_t, a_t) = r_t + \gamma Q(s_{t+1}, \underset{a}{\operatorname{argmax}}\, Q(s_{t+1}, a; \theta_t^-); \theta_t) \quad (1)$$

The Loss Function is used to compute the temporal difference error to back-propagate PN and update the model weights. The TN role is to stabilize the learning process and improve the long-term memory of the agent (via network updates every N-steps with the decay effect applied). One of the main issues of the QL is its susceptibility to overestimate the commonly taken actions beyond the realistic values [16]. The Double QL (used in DDQN), however, in the initial steps, can underestimate the values of actions but does not suffer from overestimation bias (the prior issue can be solved by setting initial high values for the actions chosen less frequently). Hence, by using weights θ_t^- to estimate Q-values, fair evaluation of the policy during each step and mitigation of action-values overestimation is possible (typical issue of vanilla DQN-based solutions).

B. TASAC system model

We model the SAC problem as the Markov Decision Process (MDP) $\mathcal{M} = (S; A; T; R)$, where S – set of states, A – set of actions a, T – transition probability from state s_t to s_{t+1} at time t after taking action a_t, R – rewards r received for actions. The resource pool available in the system is denoted by $\mathcal{R}$ (we take the value as a scalar). Each SAR $k \in \mathcal{K}$ is described by a tuple $\langle b_r^k, q^k, t_d^k, t_t^k \rangle$, which defines maximum resources requested by the slice b_r^k (measured in units u), the SST q^k ($q^k = 0$ for ToD-dependent eMBB-like slices, $q^k = 1$ for ToD-independent mMTC-like slices), NSI deployment t_d^k and termination t_t^k time (in time units tu). The state s is composed of 6 parameters: Total Requested Resources (TRR, eq. 2), Total Consumed Resources (TCR, eq. 3), Prediction of Total Consumed Resources (PTCR, eq. 4), the ratio of accepted mMTC (eq. 5) and eMBB slices (eq. 6), relative ToD (T-length of the day in tu, eq. 7), and the type of the recently requested slice q^k.

$$TRR = \sum_{k=1}^{\mathcal{K}} \frac{b_r^k(t)}{\mathcal{R}} \tag{2}$$

$$TCR = \sum_{k=1}^{\mathcal{K}} \frac{b_c^k(t)}{\mathcal{R}} \tag{3}$$

$$PTCR = \sum_{k=1}^{\mathcal{K}} \frac{b_p^k(t)}{\mathcal{R}} \tag{4}$$

$$\varphi_{mMTC} = \sum_{k=1}^{\mathcal{K}} \frac{1}{|\mathcal{K}|} \; iff \; q^k = 1 \tag{5}$$

$$\varphi_{eMBB} = \sum_{k=1}^{\mathcal{K}} \frac{1}{|\mathcal{K}|} \; iff \; q^k = 0 \tag{6}$$

$$ToD = (t \mod T)/T \tag{7}$$

It must be noted that the network state vector can be extended depending on the adopted slice classification, i.e., the operator can offer other SSTs with specific ToD activity. Resource Consumption Monitor and Resource Consumption Predictor provide information about resource consumption b_c and b_p as described in Section III-A and depicted in Fig. 4.

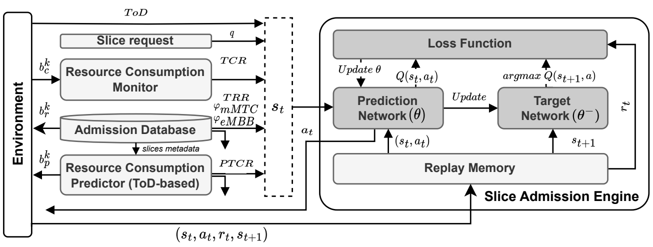


Fig. 4. Data flow in TASAC algorithm

Two actions are permitted for the agent to take, SAR acceptance $a = 1$ and SAR rejection $a = 0$. The agent's main goal is to maximize utilization in the time frame. As the future resources consumed by NSI are, in general, not known, the agent reward function (cf. eq. (8)) is based on the forecast of the consumed resources in the period of NSI runtime, i.e., $\langle t_d, t_t \rangle$ denoted as $|T|$ and measured in time units tu.

Whenever the consumed resources exceed the resource pool, the violation occurs. It has to be noted that, from the MNO point of view, the most adverse effects result from significant resource violations. The penalty assigned to the agent for overutilization is based on the maximum crossing over the resource limit. The resource prediction relies on the ToD curve in the case of eMBB, while in the case of mMTC, where ToD effect is not present, it is a constant value (i.e., the maximum resources requested in SAR).

$$r(a, t_d, t_t) = \begin{cases} 0 & \text{iff } a = 0 \text{ and (9)} \\ 0.5 & \text{iff } a = 0 \text{ and not (9)} \\ \sum_{t=t_d}^{t_t} \sum_{k=1}^{\mathcal{K}} \frac{b_r^k(t)}{\mathcal{R}|T|} & \text{iff } a = 1 \text{ and (9)} \\ 1 - \max_{t_d \le t \le t_t} b_p(t) & \text{otherwise} \end{cases} \tag{8}$$

$$\wedge_{t \in \langle t_d; t_t \rangle} \sum_{k=1}^{\mathcal{K}} \frac{b_p^k(t)}{\mathcal{R}} \le 1 \tag{9}$$

The desired final state for MNO is when the resources consumed in the analyzed time frame equal the resource pool $\mathcal{R}$. The proposed reward function provides the agent with increased rewards for maximizing the acceptance rate for the most beneficial NSIs, i.e., requesting large amounts of resources (the price proportional to peak requirements). Moreover, we reward the agent for not accepting SARs, if the predicted consumption would have exceeded the available resources.

As the baseline for comparison, we use a typical DQN implementation, which omits ToD-based resource consumption (fixed resources) and resource prediction. The state is a tuple $s = (TRR, TCR, \varphi_{mMTC}, \varphi_{eMBB}, ToD, q^k)$, and the reward function denoted in eq. 10 is used.

$$r(a, t_d, t_t) = \begin{cases} 0 & \text{iff } a = 0 \text{ and (11)} \\ 0.5 & \text{iff } a = 0 \text{ and not (11)} \\ \sum_{t=t_d}^{t_t} \sum_{k=1}^{\mathcal{K}} \frac{b_r^k(t)}{\mathcal{R}|T|} & \text{iff } a = 1 \text{ and (11)} \\ 1 - \max_{t_d \le t \le t_t} b_r(t) & \text{otherwise} \end{cases} \tag{10}$$

$$\wedge_{t \in \langle t_d; t_t \rangle} \sum_{k=1}^{\mathcal{K}} \frac{b_r^k(t)}{\mathcal{R}} \le 1 \tag{11}$$

V. Concept Evaluation

Without the loss of generalization, we focus on a connectivity resource type only, i.e., bandwidth. The TASAC concept, however, can be easily extended towards any type of resource. The TASAC algorithm has been evaluated using an event-driven Python-based simulator using SimPy [17] package. Three test scenarios have been conducted (see Table I) that aim to assess the TASAC performance gains over the traditional DQN-based SAC (T1) and its behavior under different ratios of arriving eMBB and mMTC SARs (T1, T2, T3). For each test, the same resource pool $\mathcal{R} = 3000$, sampling of $tu = 1$

TABLE I
TEST SCENARIOS

ID	λ_u	$eMBB : mMTC$	Description
T1	6	1:1	normal network load
T2	6	1:2	higher rate of ToD-independent slices
T3	6	2:1	higher rate of ToD-dependent slices

and day duration $T = 1440\ tu$ (equivalent to 1 min sampling interval) is considered.

We follow the most popular approaches and model the influx of admission requests as the Poisson process [2], [3], [5]. The requested SSTs follow a discrete uniform distribution, $\mathcal{U}\{0, 1\}$, and the maximum resources requested by slices are 10 u for mMTC NSI and 100 u for the eMBB NSI. In the evaluation, we consider short-lived slices – the duration of NSI follows the discrete uniform distribution $\mathcal{U}(30, 1440)$ – duration of 30 min to 24 hours. The ToD activity of the eMBB slice is modeled by the curve derived from the data acquired from the carrier-grade transport network using the operator's monitoring system (cf. Fig. 1). Resource consumption predictions are provided using the perfect predictor (i.e., the consumed resources are equal to the predictions). The tests are performed for the DNN architecture consisting of 4 dense layers with 16 hidden neurons each and ReLu activation function, and linear activation layer (both PN and TN). The DNN training is performed using the Huber loss function with learning rate $lr = 0.00025$ and the Adam optimizer. The TASAC/DQN-specific hyperparameters setup includes discount factor $\gamma = 0.5$, $\epsilon_{max} = 1$, $\epsilon_{min} = 0.01$, $\epsilon_{decay} = 0.999$, TN soft update coefficient $\tau = 0.99$.

A. Performance and gains over DQN

To measure the agents' performance, the average requested and consumed resources are confronted with the total penalty obtained during consecutive training episodes. The comparison of results obtained for the DQN and TASAC in the baseline scenario (T1) are presented in Fig. 5 and Fig. 6.

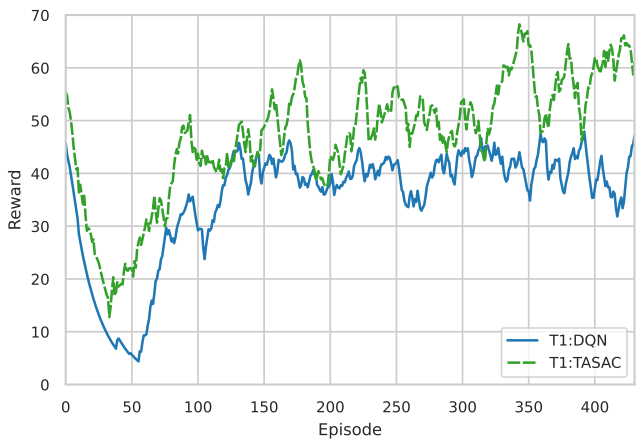


Fig. 5. Comparison of rewards obtained by DQN and TASAC agents

The TASAC approach enables the accumulation of much higher rewards from the beginning of the training process. The average resource utilization gain compared to the standard DQN approach is over 25% during the operation. Also, it has to be noted that in the analyzed case, the TASAC concept

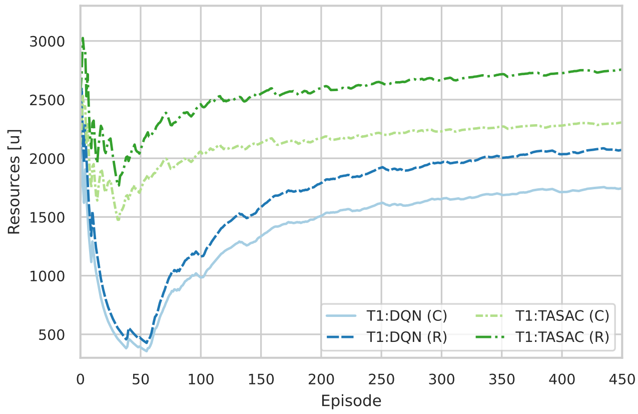


Fig. 6. Requested (R) vs. consumed (C) resources for DQN and TASAC

overestimates action-values in the initial phase of training (resource usage drop beyond $\approx 20\%\,\mathcal{R}$ for DQN around episode 50, while for TASAC the consumption is over $\approx 50\%\,\mathcal{R}$ for the entire duration of the training). Moreover, exploiting the prediction information largely improved the convergence time to the decent policy (over 100 episodes faster than the DQN-based agent). The final large disproportion of resource utilization indicates the positive impact of using the prediction information in the SAC process.

The cumulative gains of TASAC admission strategy over conventional DQN are presented in Table II. Once the admission policies stabilize (around episode 200), TASAC enables the admission of slices requesting around 60% more resources resulting in massive financial gain from the MNO perspective.

B. Performance under different SST ratios

The second test examines the performance of the TASAC algorithm under the influx of SARs with different ratios of eMBB and mMTC slices (scenario T1, T2, T3, cf. Table I). The obtained results are presented in Figs. 7 and 8.

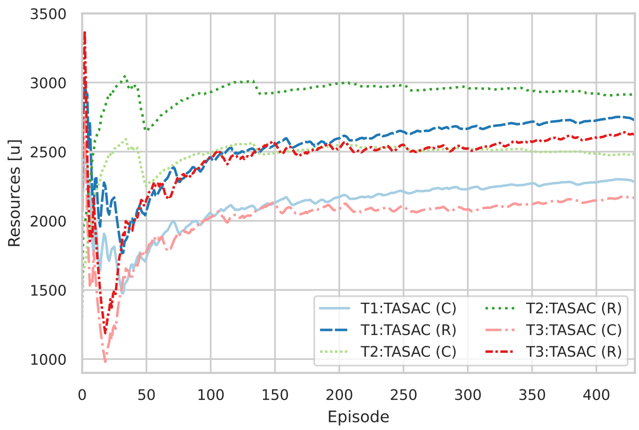


Fig. 7. Requested (R) vs. consumed (C) resources for different SST ratios

First, the best utilization was achieved when the number of mMTC SARs was the highest (T2), while the lowest utilization can be observed for the larger number of eMBB SARs (T3). The results for the latter, however, are comparable to the performance under the balanced influx of mMTC and eMBB SARs (T1), which shows that TASAC successfully selects

TABLE II
PERFORMANCE COMPARISON OF DQN VS. TASAC (CUMULATIVE)

Episode	Reward			Requested Resources			Consumed Resources		
	DQN	TASAC	Gain [%]	DQN	TASAC	Gain [%]	DQN	TASAC	Gain [%]
50	148.6	755.9	**408.7**	47662.3	107797.1	**126.2**	39718.2	90213.9	**127.1**
100	1632.3	3057.5	**87.3**	94490.8	224436.0	**137.5**	79211.4	187731.8	**137.0**
150	3793.5	5510.6	**45.3**	167948.6	350174.5	**108.5**	141231.4	292807.7	**107.3**
200	5620.9	7613.5	**35.4**	254806.1	478661.7	**87.8**	214509.7	400248.2	**86.6**
250	7642.3	10296.2	**34.7**	348340.5	609580.5	**75.0**	293353.0	509734.0	**73.8**
300	9772.1	12827.1	**31.3**	445186.9	742995.7	**66.9**	374949.1	621301.6	**65.7**
350	11927.0	15664.1	**31.3**	545034.3	877934.7	**61.1**	458988.8	734095.2	**59.9**
400	13756.1	18772.8	**36.5**	647383.5	1014162.4	**56.7**	545117.7	847979.0	**55.6**

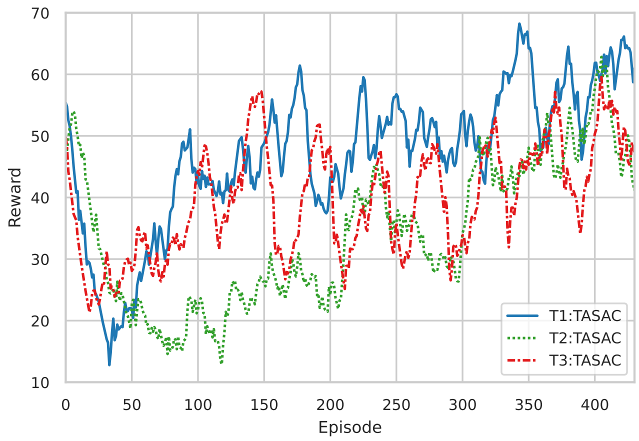


Fig. 8. Comparison of TASAC rewards obtained in scenarios T1, T2, and T3

the slices maximizing the actual utilization (i.e., mMTC). The benefit of considering ToD-dependencies can also be observed when the mMTC SST dominates – the consumption curve approaches the requested resources curve meaning more even resource utilization during each day of operation. In each of the three considered tests, TASAC has a similar policy convergence time, and successive policy improvements (increasing reward) can be observed for consecutive episodes.

VI. SUMMARY AND CONCLUSIONS

This paper presents a novel DDQN-based SAC algorithm called "TASAC". Unlike the conventional admission approaches, TASAC algorithm exploits not only the information about the maximum resources requested by the NSI tenant but also considers the actual and predicted future consumption dependent on ToD and SST. To this end, two SSTs are considered: ToD-dependent eMBB and ToD-independent mMTC. To simulate the consumption of each type, well-established data-driven models have been exploited. It has been shown that incorporating ToD-dependency and prediction can significantly improve slice admission rate and utilization of MNOs' resources. The overall gain of the TASAC algorithm comprises over 30% better cumulative reward and over 50% better resource utilization than achieved by the conventional DQN-based approach without a major increase of complexity. Moreover, the algorithm selects the slices that consume a stable amount of resources making the overall utilization more even in time. While the considerations refer only to two SSTs, the concept can be easily extended to handle additional SSTs with different activity patterns.

REFERENCES

[1] M. O. Ojijo and O. E. Falowo, "A survey on slice admission control strategies and optimization schemes in 5G network," *IEEE Access*, vol. 8, pp. 14 977–14 990, Jan. 2020, doi: 10.1109/ACCESS.2020.2967626.
[2] W. F. Villota Jácome, O. M. Caicedo Rendon, and N. L. S. da Fonseca, "Admission control for 5G network slicing based on (Deep) Reinforcement Learning," Apr. 2021, doi: 10.36227/techrxiv.14498190.
[3] S. Bakri, B. Brik, and A. Ksentini, "On using Reinforcement Learning for network slice admission control in 5G: Offline vs. online," *International Journal of Communication Systems*, vol. 34, no. 7, pp. 1–12, May 2021, doi: 10.1002/dac.4757.
[4] V. Sciancalepore, X. Costa-Perez, and A. Banchs, "RL-NSB: Reinforcement Learning-based 5G network slice broker," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1543–1557, Aug. 2019, doi: 10.1109/TNET.2019.2924471.
[5] B. Han, D. Feng, and H. D. Schotten, "A Markov model of slice admission control," *IEEE Networking Letters*, vol. 1, no. 1, pp. 2–5, Mar. 2019, doi: 10.1109/LNET.2018.2873978.
[6] D. Bega, M. Gramaglia, A. Banchs, V. Sciancalepore, and X. Costa-Pérez, "A machine learning approach to 5G infrastructure market optimization," *IEEE Transactions on Mobile Computing*, vol. 19, no. 3, pp. 498–512, Mar. 2020, doi: 10.1109/TMC.2019.2896950.
[7] M. R. Raza, C. Natalino, P. Öhlen, L. Wosinska, and P. Monti, "Reinforcement Learning for slicing in a 5G flexible RAN," *Journal of Lightwave Technology*, vol. 37, no. 20, pp. 5161–5169, Oct. 2019, doi: 10.1109/JLT.2019.2924345.
[8] D. Chmieliauskas and D. Gursnys, "LTE Cell Traffic Grow and Congestion Forecasting," 04 2019, pp. 1–5.
[9] H. D. Trinh, N. Bui, J. Widmer, L. Giupponi, and P. Dini, "Analysis and modeling of mobile traffic using real traces," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. Montreal, QC: IEEE, Oct. 2017, pp. 1–6, doi: 10.1109/PIMRC.2017.8292200.
[10] S. Wang, X. Zhang, J. Zhang, J. Feng, W. Wang, and K. Xin, "An approach for spatial-temporal traffic modeling in mobile cellular networks," *2015 27th International Teletraffic Congress*, pp. 203–209, Sep. 2015, doi: 10.1109/ITC.2015.31.
[11] V. Chetlapalli, K. S. S. Iyer, and H. Agrawal, "Modelling time-dependent aggregate traffic in 5G networks," *Telecommunication Systems*, vol. 73, no. 4, pp. 557–575, Apr. 2020, doi: 10.1007/s11235-019-00629-w.
[12] GSMA, "Generic Network Slice Template," GSM Association, Official Document NG.116, ver. 7.0, Jun. 2022. [Online]. Available: https://www.gsma.com/newsroom/wp-content/uploads/NG.116-v7.0.pdf
[13] H. van Hasselt, A. Guez, and D. Silver, "Deep Reinforcement Learning with Double Q-learning," 2015, doi: 10.48550/ARXIV.1509.06461.
[14] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with Deep Reinforcement Learning," 2013, doi: 10.48550/ARXIV.1312.5602.
[15] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, pp. 279–292, May 1992, doi: 10.1007/BF00992698.
[16] H. van Hasselt, "Double Q-learning," in *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada*, J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, Eds., vol. 3. Curran Associates, Inc., 2010, pp. 2613–2621.
[17] SimPy Project. [Online]. Available: https://pypi.org/project/simpy/

drones

Review

Advanced Air Mobility and Evolution of Mobile Networks

Lechosław Tomaszewski [1,*] and Robert Kołakowski [1,2,*]

[1] Orange Polska, 02-326 Warsaw, Poland
[2] Division of Cybersecurity, Institute of Telecommunications, Faculty of Electronics and Information Technology, Warsaw University of Technology, 00-661 Warsaw, Poland
* Correspondence: lechoslaw.tomaszewski@orange.com (L.T.); robert.kolakowski@orange.com (R.K.)

Abstract: Advanced Air Mobility (AAM) is a promising field of services based on Unmanned Aerial Vehicles (UAVs), which aims to provide people and cargo transportation services in underserved areas. The recent advancements in the fields of aviation and mobile telecommunication networks have opened up multiple opportunities for the development of disruptive AAM applications. This paper presents the overview and identifies the major requirements of emerging AAM use cases to confront them with the features provided by the 5G System (5GS), which is commonly considered the key enabler in providing commercial AAM services. The major benefits, gaps, and issues regarding using 5GS to serve AAM operations are identified and discussed. Finally, the future perspectives for AAM services are outlined with a focus on the potential benefit that can be provided as the mobile network evolves towards 6G.

Keywords: AAM; 5G; 6G; UTM; 3GPP; UAS; UAV; drone; AI; mobile networks; mobile services; requirements; architecture; applications

check for updates

Citation: Tomaszewski, L.; Kołakowski, R. Advanced Air Mobility and Evolution of Mobile Networks. *Drones* **2023**, *7*, 556. https://doi.org/10.3390/drones7090556

Academic Editors: Georgi Georgiev, Friedrich-Wilhelm Bauer, Ralf Sindelar, Diego González-Aguilera and Pablo Rodríguez-Gonzálvez

Received: 30 June 2023
Revised: 7 August 2023
Accepted: 18 August 2023
Published: 29 August 2023

1. Introduction

The Advanced Air Mobility (AAM) is an emerging class of Unmanned Aerial Vehicle (UAV)-based services focused on the transportation of individuals and cargo. With the recent advancements in the field of technical solutions related to the AAM use cases, e.g., construction of electric Vertical Take-Off and Landing (VTOL) or solid automated control, the implementation of the carrier-grade AAM services has become more feasible. This resulted in the increased rise of interest in both academia and the telecommunication and aviation business. One of the candidate telecommunications solutions commonly mentioned in the context of support and leveraging such services is the 5G System (5GS).

The goal of this review paper is first to analyse the key requirements, needs and future expectations of the AAM verticals regarding the performance, capabilities and mechanisms of the mobile telecommunications system. Based on the analysis, the verification of 5GS and research community support for AAM, together with the identification of potential gaps, is conducted. Finally, a high-level vision of potential AAM services improvements that can come with the wide deployment of the future 6G System (6GS) is outlined.

The structure of the paper is as follows. In Section 2, the AAM ecosystems characteristics are discussed with a focus on specific regulatory conditions, service types, needs and requirements that have to be satisfied to allow AAM services at the carrier-grade level. Section 3 presents the current progress of standardisation activities related to the aviation ecosystem with emphasis on integration and interactions between the Unmanned Aircraft Systems Traffic Management (UTM) system and 5GS, as well as shows the high-level view on the research activities related to the mobile networks extensions towards AAM operations. In Section 4, the key 5GS features that not only support but also leverage the AAM operations are categorised and listed. Next, the envisioned evolution of mobile networks towards 6G is described, emphasising the key elements that can further contribute to future AAM operations (cf. Section 5). Section 6 provides the verification of the feasibility of mobile network based AAM services, as use-case specific requirements are confronted with the 5GS and 6GS performance Key Performance Indicators (KPI). Finally, Section 7 summarises and concludes the paper.

2. Advanced Air Mobility Ecosystem—Conditions, Services, Needs and Requirements

Advances in multiple areas, i.a., systems of electric propulsion, electric VTOL, positioning, Artificial Intelligence (AI)-based automated control, Information and Communications Technology (ICT), and sensors, have enabled the development of the new, disruptive AAM vision, which is, according to the National Aeronautics and Space Administration (NASA) of USA, "an air transportation system that moves people and cargo between places previously not served or underserved by aviation–local, regional, intraregional, urban–using revolutionary new aircraft that are only just now becoming possible" [1] and includes both manned and pilot-supervised/autonomous unmanned aircrafts of various size and mission. AAM covers miscellaneous groups of stakeholders (commercial/industrial, public sector, military, private citizens), deployment regions (e.g., megacities/agglomerations, coastal/bay/delta, island chains, rural areas, mountainous regions), and missions (moving people, moving goods, data collection, task/process automation) [2]. European Union Aviation Safety (EASA) still limits its view to the area of Urban Air Mobility (UAM), i.e., "new safe, secure and more sustainable air transportation system for passengers and cargo in urban environments, enabled by new technologies and integrated into multimodal transportation systems" [3]. Both AAM and its subset—UAM have to face multiple requirements associated with safety, security, resilience, regulation, scalability, flexibility, and environmental impact, including sustainability.

AAM will be deployed within the UTM ecosystem [4] (U-space in the European Union (EU) [5]) composed of multiple services to support the aerial operations the binder and foundation of which is the ubiquitous ICT environment. However, the component communication network cannot only be a transparent relay of data, but it must participate in the dynamics of UTM process interactions by means of additional exposed services. Mobile network is a natural candidate for this role, as it provides ubiquity (at least potentially), unified Internet Protocol (IP)-based connectivity for various use cases and purposes, globally recognised standard of network and User Equipment (UE) implemented worldwide and providing the support for roaming, i.e., operation outside a country of UE registration, which is of premium importance in EU due to the principles of "single European sky" and "free movement of goods and services".

The four aforementioned groups of AAM missions can be split into various use cases:

- human mobility—air-TAXI, recreational/tourist/sightseeing flights (singly or in swarms), medical transport (air-ambulances), mobility of public personnel (including public security services);
- goods mobility—cargo transport, parcel delivery, medical transport (medicines, blood, organs for transplantation, samples for analysis, etc.), food transport, transport of smaller animals (e.g., to a veterinary clinic and back);
- data collection—monitoring of environmental parameters (temperature, humidity, air quality, etc.), acquisition of data from distant off-line sensors, acquisition of still images, mobile network data acquisition/3D coverage survey or monitoring, coverage and interference surveys for Industrial, Scientific, Medical (ISM) bands communication technologies, e.g., Long Range Wide Area Network (LoRaWAN), biodiversity monitoring (animals, plants, environmental changes), surveys of exhaust fumes emitted by building chimneys and detection of burning prohibited substances;
- task/process automation—photogrammetry (e.g., for geodesy/cartography, agriculture, forestry, inventory of buildings, and inventory of flora for biodiversity monitoring, disaster detection), infrastructure inspection, equipment tracking, surveillance (criminals identification and chasing, pandemic management via identification of the infected people, etc.), situational awareness.

The wireless connectivity will primarily support the needs associated with aviation itself ("non-payload" transmission): Command and Control (C2), UTM, and inter-aircraft communication (e.g., position/trajectory broadcasting or collision avoidance) as well as First Person View (FPV) for remote pilot in case of non-autonomous flights. Additionally, depending on the flight use case, various needs for "payload" connectivity may apply:

- WiFi for passengers through an aggregated mobile access of an AAM aircraft;
- Augmented Reality (AR)/Virtual Reality (VR) for sightseeing flights passengers, medical personnel or public (security) services;
- intra-swarm multimedia communication for passengers;
- cargo tracking;
- Real-Time (RT) monitoring of patients' vital signs in air ambulances;
- RT monitoring of sensitive cargo (medical transport, food, live animals, special shipments)—both using sensors (temperature, pressure, etc.) and multimedia streaming;
- still images acquisition associated with geofencing (for imaging-restriction areas);
- acquisition of diverse geo-tagged data with on-board sensors;
- relaying of stationary off-board sensors data;
- RT multimedia streaming from a remotely controlled audio/video acquisition system for remote inspection/awareness.

Future AAM aircraft can include devices supporting a single purpose of mission or multiple purposes. Such multi-purpose AAM devices, along with technological development, can evolve toward programmable ICT platforms, and new AAM business models can emerge, e.g., "AAM-ridesharing" (analogous to "ridesharing" or "carpooling" in the automotive field)—the AAM operators can offer "side effect" services based on data additionally acquired during multiple, other purpose flights (cargo/parcel delivery, air-TAXI, etc.): crowdsensing of environmental parameters, creation of 3D mobile network coverage maps or spatial visualisation of individual base station effective radiation characteristics, detecting the presence of certain species of wild animals in urban areas, and even offering surveys/measurements/inspections at a specific place and time based on a scheduled flight plan. Such "AAM-ridesharing" approach would also be justified in terms of environmental impact or sustainability.

The fundamental difference between AAM and terrestrial mobility is that the former creates a nodal transportation network (direct connections between any points), and the latter creates a linear network along permanent communication routes limited by geography. AAM provides flexible capacity, inherent adaptability to flow changes, resilience to disruptions, and protection against congestion [2]. However, all these positive features for transport flow pose a challenge to the communication platform, which must provide support anywhere, anytime, and often without prior notice.

3. Related Work

3rd Generation Partnership Project (3GPP) has been dealing methodically with the problem of Unmanned Aviation System (UAS) support by the 5G Public Land Mobile Network (PLMN) since 2018. The alliance follows the 3-stage methodology of International Telecommunication Union (ITU) composed of: Stage 1—service requirements from the user point of view, Stage 2—architecture to support the service requirements, and Stage 3—details of the architecture implementation. 3GPP issues technical reports (non-normative) and technical standards (normative). The first 3GPP release standardising 5GS was the 3GPP Release 15, and the first one to deal with UAS support was the 3GPP Release 16. According to the roadmap (cf. Figure 1, the 3GPP Release 17 is frozen since the middle of 2022, but its fundamental documents have been still under the change control (change requests being processed) until the last session in June 2023 [6]. Currently, according to the timeline, the 3GPP Release 18 Stage 2 has just been frozen, while its Stage 3 will be frozen at the end of 2023. The studies on the scope of the future 3GPP Release 19 are in progress, its scope definition will be concluded also at the end of 2023, and it is expected to be the first one dealing with the 6GS requirements. The 3GPP Release 20 will probably perform first 6GS studies, and the 3GPP Release 21 will likely bring the first "6G Basic" drop in late 2028 (indicative dates, assuming the typical 3GPP release cycle) [7].

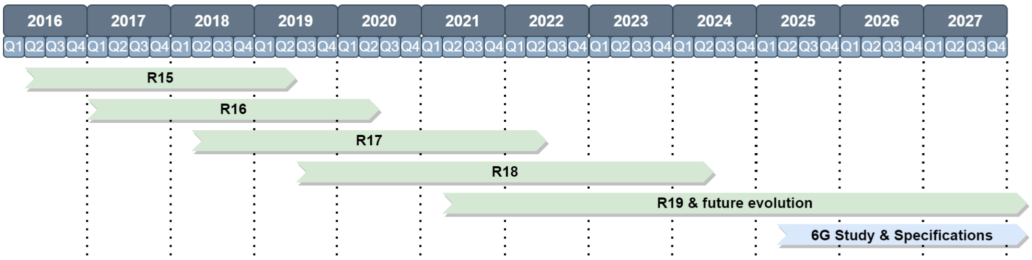


Figure 1. 3GPP Releases timeline (based on [7–9])—green: 5G (Relase 15–Release 17) and 5G Advanced (since Release 18), blue: 6G.

In the initial study [10], ten use cases have been described: initial UAS authorisation to operate, live data acquisition by UTM, data acquisition by law enforcement, enforcement of no-fly zones for UAV, distributed close-field separation service for mutual Unmanned Aerial Vehicles (UAVs) collision avoidance, local broadcast of UAS identity, differentiation between UAV-specific UE and regular UE attached to UAV, cloud-based Beyond Visual Line of Sight (BVLOS) UAV operation, UAV fly range restriction, and UAS-based remote inspection. On this basis, 3GPP has defined a separate normative Stage 1 document [11] specifically referring to UAS service requirements for 5GS, both functional and Quality of Service (QoS) ones. In the successive study [12], the description, gap analysis vs. the Stage 1 standard [11], and potential new requirements identification have been performed for additional 13 use cases, namely: UAV supporting high resolution video live broadcast application, radio access node on-board UAV, UAS C2 communication, simultaneous support for UAVs and Enhanced Mobile Broadband (eMBB) users data transmission, autonomous UAVs controlled by AI, isolated deployment of radio access through UAV, radio access through UAV, separation of UAV service area from other User Equipments (UEs) served by a base station, UAV communication service experience assurance, service availability to UAVs swarm of UAVs in logistics, changing UAV Ground Control Station (GCS), and framework for steering KPIs of UAV. Within the scope of the 3GPP Release 19, there is also an on-going study of additional capabilities of PLMN for drone operations and management (report in maturing phase) [13]. The study defines and analyses the new set of 7 use cases, including identification of their new requirements, namely: detection of UE on-board UAV, supporting UAV pre-flight preparation (PLMN to answer flight feasibility query by UTM), geofencing for Visual Line of Sight (VLOS) UAV missions, network-assisted UAV Detect And Avoid (DAA)—detection and collision avoidance (relative distance between UAVs to be determined and reported to UTM by PLMN), 3GPP network as an information (spatial, surveillance, meteorological, and RT positioning information) source for UTM, support for UAV in-flight operations with RT reporting of QoS, performance, and stability of C2 link along the route, and UAV flight route tracking at rendez-vous points (PLMN as a UAV traffic sensor and locator, also in the case of non-networked UAVs or served by other Public Land Mobile Networks (PLMNs) in the area).

Table 1 presents example 5GS performance requirements for UAS support according to 3GPP [11]. 3GPP recognises the following C2 modes: steer to waypoints, direct stick steering, automatic flight on UTM, and approaching autonomous navigation infrastructure. It is additionally stated that the C2 communication between UAS and UTM can be allowed to experience much longer traffic interruptions at the application layer, e.g., timeouts of 30 s on the uplink and 300 s on the downlink. The data rate requirements for these C2 modes have not been explicitly defined, but they can be derived from the given message sizes at specified latencies. The most challenging data rate 3.2 Mbps is for the last C2 mode. Examples of non-C2 communication services for UAS show a significant scatter of performance requirements. For the communication from UTM to UAS, the data rate is not specified, but it can be derived as above—the estimated value is 16 kbps. The video quality assumed by 3GPP for Beyond Visual Line of Sight (BVLOS) FPV is not high. Peak performance requirements for UAS are for the 4 × 4 K AI surveillance communication service.

Table 1. Selected 5GS performance requirements for UAS support according to 3GPP [11].

C2 Mode	E2E Latency	Reliability	Maximum UAV Ground Speed
Steer to waypoints	1 s	99.9%	300 km/h
Direct stick steering	40 ms	99.9%	60 km/h
Automatic flight on UTM	5 s	99.9%	300 km/h
Approaching autonomous navigation infrastructure	10 ms (downlink) 140 ms (uplink)	99% (downlink) 99.99% (uplink)	50 km/h
Example UAS communication service	**E2E latency**	**Reliability**	**Data rate**
UTM to UAS communication	500 ms	99.9%	Not specified
BVLOS FPV video	140 ms	99.99%	4 Mbps (uplink) at 720 p, 30 fps
4 × 4 K AI surveillance	20 ms	Not specified	50 Mbps (downlink) 120 Mbps (uplink)

Stage 2 work began with studies on supporting UAS connectivity, identification and tracking [14] and application layer support for UAS [15]. The former dealt with solutions to the network-level key issues: UAV and its GCS identification, UAV authorisation by UTM, UAV GCS identification and authorisation/authentication, UAV and its GCS tracking, UAV authorisation revocation and (re)authorisation failures, UAV GCS and UAV(s) association, and connectivity to UTM for UAV. The latter concerned solutions to the application-level key issues: broadcast communications, UAV location information reporting to UTM, exposure of capability to access UAS-related information, UAV application server QoS provisioning, switching and selecting C2 communication modes, UTM information provisioning to UAV and/or its GCS, UAS identification capability usage (also in case of UAS components change/update) in the application layer, media session monitoring and management, the impact of UAS application hosting in an edge data network instead of cloud, support for reporting of RT UAV UE monitoring information, and tracking of UAV location deviation. The proposed solutions from both studies have been analysed, and the selected solutions have been put into the Stage 2 normative documents, respectively network-level [16] and application-level [17] ones. It is worth a mention that the fundamental Stage 2 outcomes include the identification of integration architecture of PLMN and the UTM/AAM ecosystem (cf. Figure 2), definition of mechanisms of joint PLMN and UTM authentication and authorisation of a UAV to operate as well as UAS C2 connectivity control, UAVs position tracking and reporting, and defining a special Unmanned Aviation System Network Function (UAS NF) whose task is to act as a standardised mediator between PLMN and the UTM/UAS ecosystem and expose all 5GS control mechanisms and information related to UAS communication support. In the recently completed studies, further network-level [18] and application-level [19] key issues associated with UAS support have been analysed and solutions have been proposed for further selection and inclusion into the standardisation. The former, addressing further architecture enhancements for Unmanned Aviation Systems (UASs) and explicitly UAM, has been dealing with communication between UAVs or UAV and its GCS through a direct 5G inter-UE link bypassing PLMN (base stations and core), short-range UAV identifier broadcasting, and support for short-range area DAA based on direct inter-UAV links. The latter has addressed the application-level issues of a direct inter-UE communication between UAVs or UAV and its GCS, support for a UTM change during a flight and for DAA.

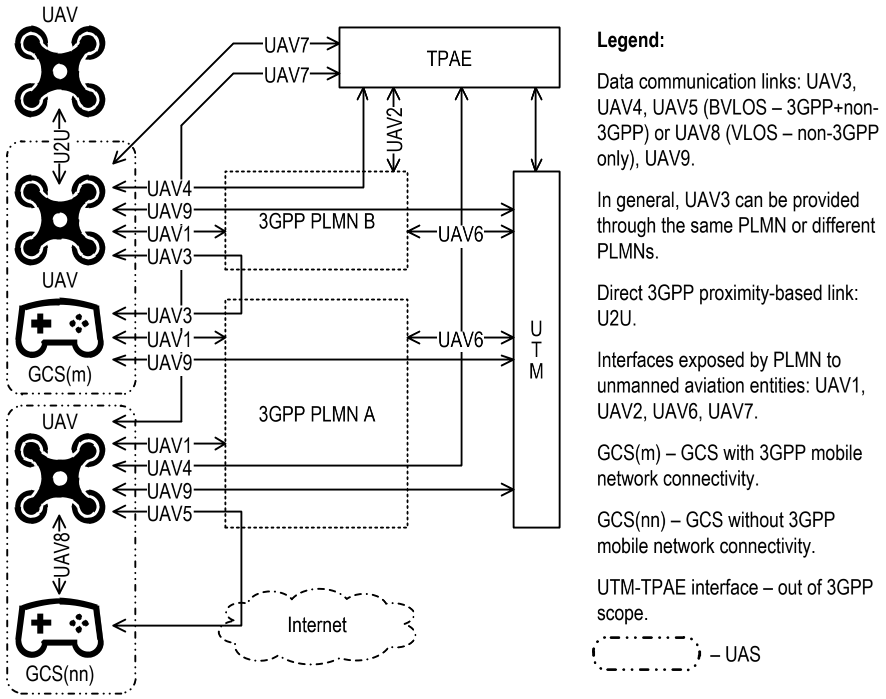


Figure 2. Reference architecture of the UTM ecosystem and mobile network integration (based on [14]).

Additionally, the security aspects of UAS have been studied and standardised. The initial study has analysed the threats, potential security requirements, and proposed solutions of the following key issues: UAS authentication and authorisation, pairing authorisation for UAV and its GCS, Third-Party Authorised Entity (TPAE) authentication and authorisation, location information veracity and location tracking authorisation, privacy protection of UAS identities, security protection of information in remote identification and between UAV/GCS and UTM, security of C2 communication [20]. The study outcomes have fed the normative document scope, namely authorisation and pairing of UAS components as well as location tracking authorisation and veracity [21]. There is another completed Release 18 study on security aspects of UAS, UAV, and UAM, dedicated to direct inter-UE C2 security, authorisation and privacy, security and privacy of DAA unicast and broadcast connection, and UAV identifier broadcast privacy and security [22].

The 3GPP Stage 3 documents associated with the UAS support include definitions of services offered and Application Programming Interfaces (APIs) of UAS NF [23] and of the UTM mediation module exposed toward PLMN and embedded in it to support the joint PLMN and UTM authentication and authorisation of UAVs [24].

The progress summary of 5GS support dedicated for Unmanned Aviation Systems (UASs) in specific 3GPP releases is provided in Table 2. The status of support depends on the individual release:

- Release 17: included in standardisation (release completed).
- Release 18: work in progress.
- Release 19: scope candidate use cases waiting for approval at the end of 2023.

No dedicated features specifically supporting UAS in Release 15 and Release 16.

Table 2. Summary of dedicated UAS support roadmap in 3GPP releases (based on [13,18,19,25,26], features' availability details in the text).

Feature	Release ID
UAV remote identification Joint PLMN and UTM authentication and authorisation of UAV Authorisation of C2 communication UAV location reporting and tracking Communication between UAVs within a defined proximity area Pairing of UAV with its GCS (also cross-PLMN) Provisioning of C2 communication QoS for UAS C2 communication mode selection and switching RT UAV connection status monitoring and location reporting	Release 17
Direct UAV–UAV or UAV–GCS communication over direct UE–UE connection Support of UAV interaction with multiple UTM during a flight Support of DAA services Transport of C2 communication over direct UE–UE connection Broadcasting of UAV remote identification over direct UE–UE connection Support of DAA mechanism over direct UE–UE connection Enhancements of signal measurement reporting by UAV UE to 5G base station considering UAV specificity (including reporting of altitude, location, and speed) Flight path reporting by UAV to 5G base station for radio resource planning aid Study of beamforming support at UAV UE side (in 410–7125 MHz range)	Release 18
Detection of UE with UAV-specific additional features Responding flight feasibility query from UTM by PLMN Geofencing for VLOS UAV missions DAA support by PLMN (reporting relative distance between UAVs to UTM) PLMN as location, identity, and sensing information source on UTM request Support for UAV in-flight operations with RT monitoring of C2 link along the route PLMN as a sensor of aerial traffic of UAVs for UTM	Release 19

While 3GPP is an industrial standardisation alliance, it is also fed by the contributions resulting from scientific research activities led by the alliance partners, in particular as part of research and innovation programs of EU and its bodies, namely Horizon 2020/Horizon Europe or aviation-oriented EU partnership programmes such as Single European Sky ATM Reaserch (SESAR) Joint Undertaking. The Horizon 2020 5G!Drones [27] project has integrated the domains of aviation (traffic management, drone control), and the 5G network to conduct trials of various AAM scenarios, including data collection (UAV-enhanced Internet of Things (IoT) data acquisition, environmental mapping, 5G QoS 3D mapping, long-range powerline inspection, suspicious activity detection), task and processes automation (wildfire monitoring, disaster recovery), and goods mobility (medicine delivery). The project has also provided the validation of KPIs for 5G-UAS services, proposals of mechanisms for joint 5GS–UTM authorisation of UAVs with required architectural extensions [28], and a roaming 5G architecture enabling service continuity during the cross-border UAS operations [29], without trialling handovers, however. The EU-Korean collaborative project PriMO-5G [30], analysed the use of 5G-based communication to enhance situational awareness to support firefighting operations in forest and urban scenarios. Another Horizon project ETHER aims to provide a multilayered 3D flexible and sustainable unified Radio Access Network (RAN) architecture, air interface technical innovations and data-driven automated resources management to enable the integration of Terrestrial Network (TN) and Non-Terrestrial Network (NTN). Part of the project's interest is to demonstrate the implemented architectural support (including horizontal and vertical handovers) for airspace safety-critical operations. In the SESAR GOF 2.0 project [31], the highly automated unified Air Traffic Management (ATM) has as been demonstrated in the use cases related to dense operations in control zones, airspaces mobility (controlled/uncontrolled airspace and U-space airspace/non-U-space) and cross-border operations. Also, the key enablers for cooperation between telecom and UTM ecosystem facilitating AAM operations have been investigated, which included the interfaces and data models for automated data exchange or mobile-network-based population density prediction systems [32]. The major activities in the field of AAM are also undertaken outside of EU. The Open Generation 5G Consortium targets AAM use cases (commercial parcel delivery, static infrastructure inspection, emergency response, and indoor inspection and security via UAS) that can highly benefit from 5G-based communication and aim to implement testbeds, experimentation and demonstrations to validate the scenarios [33].

GSM Association (GSMA), an association of Mobile Network Operators (MNOs) worldwide, and Global UTM Association (GUTMA) have established the Aerial Connectivity Joint Activity (ACJA) initiative to build mutual understanding and cooperation between the unmanned aviation and mobile network domains. The outcomes of the group [34] focus on the definition of interfaces for data exchange between both industries' ICT ecosystems and managerial services exposed by PLMN towards the UTM system. The ACJA documents do not define the requirements for communication services as such to be consumed by the UAS domain.

4. Key 5GS Support for AAM Operations

The 5G technology is expected to revolutionise the industry by providing a flexible, reliable and highly efficient communication platform adaptable to specific use cases as well as network extensions to unleash the full potential of vertical services. Apart from the generic mechanisms that can be exploited by any branch of industry, 5GS offers very specific support to the UAS ecosystem enabling the accommodation of the UAV-based services including AAM. Some of the key 5GS paradigms and features that can bring substantial benefits and leverage AAM services include:

- "All-IP"—Separation of network access layer and services layer. The former is based on the unified IP packet network. Services are built upon the IP-based access. The approach has been introduced to mobile networks with 4G. However, to enable very low power consumption devices, User Equipments (UEs) using only the services based on network signalling without implementation of IP stack for user data, are also supported in 5GS.
- Communication network softwarisation—Formerly, the mobile network used to be implemented with monolithic telecom infrastructure using proprietary, application-specific hardware (HW) and closed interfaces. The fundamental assumption of 5GS is softwarisation of the communication network architecture functional entities using the commodity HW. Consequently, it is possible to achieve flexibility, scalability, (re-)programmability, and a high degree of automation of the network.
- Virtualisation—Decoupling of software (SW) and HW by a HW virtualisation layer. SW is not run on a dedicated HW but on virtualised resources exposed to it (also dynamically scaled), based on a pool of physical resources behind the virtualisation layer. SW and HW life cycles are therefore separated; it is possible, e.g., to perform expansion or maintenance activities of HW without impact on SW.
- High modularity—The 5GS architecture has been modularised with high granularity, which serves to implement exactly the functionalities that are needed instead of occupying resources with largely unused ones. In this way, network functions can be individually placed as close as possible to the focal points of demand and also individually scaled. An additional factor is the Control and User Plane Separation (CUPS) principle, thanks to which the handling of control traffic in Control Plane (CP) and user traffic in User Plane (UP) can be treated and optimised separately.
- Service-Based Architecture (SBA)—The 5GS Core Network (CN) CP is designed as a set of interconnected Network Functions (NFs) producing and consuming services according to "subscribe-notify" and "request-response" mechanisms upon the RESTful APIs, JSON serialisation, and HTTP/2. The inter-Network Function (NF) communication in CP is provided and managed by a message broker, which also allows to

logically partition CP or manage the access of individual CP Network Functions (NFs) to each other.

- Network exposure—The network control mechanisms of CP are to be exposed to the external third-party systems, e.g., vertical industries' ICT environments. Additionally, the "embassies" of trusted systems (some functional modules, e.g., for mediation with PLMN) can be directly embedded in CP due to the concept of generic Application Function (AF) interacting with the native CP NFs according to SBA mechanisms.
- Network Slicing (NS)—The implementation of the "all-IP" 4G network revealed the problem that the requirements of different services with sometimes extremely different specificities are irreconcilable in the unified UP architecture, which in consequence does not provide satisfactory services support by PLMN. Therefore, the NS approach was proposed in which certain common CP mechanisms (e.g., mobility management, user data session control, user capabilities subscription management, network capabilities exposure, etc.) can be extended, using SBA ones, with some service- or application-specific features, and the user traffic can be treated separately in a UP chain architecture adapted specifically to the service or application. The atomic functions to compose the UP traffic handling chain can involve deep packet inspection, traffic forwarding or redirection, flow duplication and parallelisation to improve reliability, packet selective marking, enriching or altering, packet classification and encapsulation, firewall, anti-virus protection, parental control, etc. This way, PLMN evolves from a universal network to a "federation" of virtual parallel networks ("network slices") over a shared infrastructure and common control mechanisms. NS is also supported by QoS control and traffic prioritisation mechanisms. The wide implementation of NS, however, is significantly delayed by its technical complexity as well as the legislative implications of the regulations related to the "network neutrality" concept. The latter can potentially limit the deployment of NS-based services over the public network to mitigate their potential impact on consumers of internet access services [35].
- Management and Orchestration (MANO)—The 3GPP-compliant European Telecommunications Standards Institute (ETSI) MANO orchestrator [36] provides the network operators with the mechanisms to dynamically orchestrate and manage network slices with their components—the functionalities of critical importance in the context of infrastructural resources usage optimisations, slices' Service Level Agreement (SLA) maintenance and adaption of the deployed slices to the network workloads, etc. From the business perspective, the ability to deploy network slices and expose relevant management interfaces to the tenants in a fast manner positively contributes to both the cost as well as Time To Market (TTM) for the vertical customers' services.
- Direct inter-UE communication—The capability of direct communication between UEs in proximity. The same radio interface is used as in the case of the link between UE and the base station. The cross-PLMN communication is also assumed (exchange between UEs attached to different PLMNs).
- Network performance—5GS is designed to offer a significant communication performance that includes peak data rates of 20 Gbit/s in the downlink and 10 Gbit/s in the uplink, user-experienced data rates up to 100 Mbit/s, close to 1 ms latencies in the radio access layer, area traffic capacity of 10 Mbit/s/m^2, support for up to 1 million devices per km^2, and top spectral efficiency of 30 bit/s/Hz. The network also should support objects' mobility up to 500 km/h with zero interruption time, which should be sufficient for the vast majority of AAM use cases [37,38].

The standalone 5GS architectural framework, specified by 3GPP based on the above principles and paradigms, is presented in Figure 3. Its features indicated in the diagram include [39]:

- SBA-based CP, using message broker—Service Communication Proxy (SCP) for signalling exchange to which all other CP NFs are connected (discussed below, shown in Figure 3 as individual NFs or grouped functionally for convenience); Network Repository Function (NRF) provides the connected NFs repository to be used, e.g., during a CP service producer discovery by a consumer NF.

- Access and mobility management—both RAN with base stations and the Access and Mobility Management Function (AMF) entity, which acts as a UE and RAN nodes signalling termination point and their entry (gateway) to CP mechanisms. Particularly, AMF manages the UE mobility mechanisms, e.g., handovers.
- User traffic handling—composed by global Policy Control Function (PCF) and pairs of User Plane Function (UPF) and its own Session Management Function (SMF)—both separated according to CUPS. UPF, tailored to service specificity, is terminated in a specific Data Network (DN), e.g., public Internet, non-public IP network, etc. PCF controls, e.g., UE QoS) according to the user subscription.
- Authentication and authorisation—of both customers—Authentication Server Function (AUSF) and UE—5G Equipment Identity Register (5G-EIR): stolen terminals, terminals' capabilities, etc.
- Data management—group of NFs providing general user and network data management and storage mechanisms served commonly to various CP NFs: Unified Data Management (UDM), Unified Data Repository (UDR), and Unstructured Data Storage Function (UDSF).
- Network slicing—NFs that may include such network slicing-supporting entities like: Network Slice Selection Function (NSSF)—used during slice request by UE, Network Slice-specific and Stand-alone non-public network Authentication and Authorisation Function (NSSAAF)—slice-/application-specific authentication and authorisation mechanisms, and Network Slice Admission Control Function (NSACF)—UE admission control mechanisms. These NFs together with SMF/UPF pair constitute a network slice instance.
- Network data analytics—mechanisms of network events/performance data collection, mediation, storage, and analytics: Data Collection Coordination Function (DCCF), Messaging Framework Adaptor Function (MFAF), Analytics Data Repository Function (ADRF), and NetWork Data Analytics Function (NWDAF), respectively.
- Location—extension of 5GS CP providing a network- and UE-assisted LoCation Services (LCS) framework [40].
- Edge computing/Multi-access Edge Computing (MEC) support—functions for integration of 5GS with edge computing frameworks [41], including ETSI MEC.
- Short Message Service Function (SMSF)—service management subscription data checking and charging, messages relaying and delivering.
- AF—application part embedded in 5GS CP to interact with 3GPP CN in order to provide services.
- Network Exposure Function (NEF)—entity acting as a CP gateway for external systems and AFs non-native for 5GS CP, but embedded therein. It offers, i.a., exposure of capabilities and data, secure provision of information from the external system to 3GPP network, and translation of exchanged internal-external information.
- UAS NF—functionality supported by NEF for external exposure of CP services and data to UTM [16]. It may be either implemented as a specialised NEF or embedded in a multi-purpose NEF.

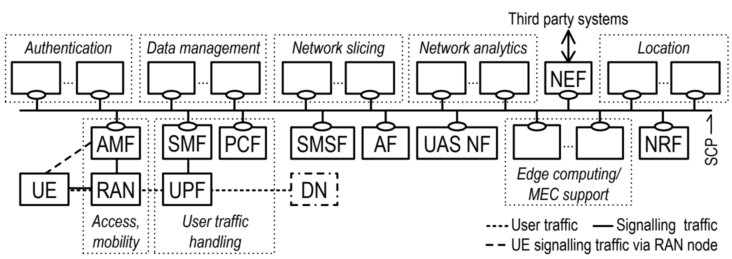


Figure 3. 3GPP 5GS architectural framework—standalone variant (based on [39]).

To leverage softwarised character, exploit modularity, and increase the openness of 5GS to the external stakeholders, several generic solutions have been proposed to facilitate the 5GS cooperation and data exchange with external systems. The support for integration, noteworthy in the AAM context, include so far:

- Vertical applications—The implementation of 5GS functions' northbound interfaces follows the common unified rules specified within, i.a., Common API Framework for 3GPP northbound APIs (CAPIF) [42], which allows secure APIs exposure in two ways (5GS to the third parties, and third-party APIs to the 5GS CP). The CAPIF framework also unifies the mechanisms of AFs on-boarding/off-boarding, service discovery, service management, event subscriptions, security and charging aspects. The other mechanism that leverages vertical applications development, deployment and integration is the Service Enabler Architecture Layer (SEAL) framework, which exposes common service enablers and CAPIF-compliant APIs for verticals. The currently standardised services include the management of groups (creation of groups of devices, e.g., for group communication), location (providing device location from 3GPP and non-3GPP source), identity and key (authentication, authorisation of users and devices), and network resources (application-specific usage and monitoring of resources utilisation by devices) [43]. The SEAL framework also supports distributed deployments and inter-service communication between different SEAL servers for improved spatial availability (e.g., for the purpose of location-based group management).
- Edge environments—3GPP proposes specialised CN and UE functions as well as relevant interfaces to facilitate the interactions of clients with the applications deployed at the network edge. The exposed capabilities include application servers discovery, provisioning of information regarding registration, configuration and connection to the edge services, support for service continuity, charging, exposure of 5GS CP mechanisms, etc. [44]. One of the most promising edge solutions, interworking with 5GS, is MEC. In addition to providing the ecosystem for management and orchestration of edge applications, MEC exposes several APIs to leverage vertical services, which include, i.a., Radio Network Information [45] (PLMN information, radio network measurements), Location (device location, information on distance to specific location or device, zonal presence) [46], UE Identity (identifiers allowing UE specific traffic rules in the MEC system) [47] or Traffic Management [48] (per application session data rate management) API. It has to be emphasised, however, that 3GPP provides a generic solution to enable the application-level integration with any third-party edge solution to provide openness and a high degree of flexibility of deployment for vertical edge applications.
- UTM ecosystem—The interconnection between the 5GS-UTM ecosystem is implemented by the specialised 5GS CP function called UAS NF. UAS NF plays the role of a mediator between 5GS and UTM by facilitating data exchange between entities as well as joint operations. It supports the mechanisms for remote identification, UAV authentication and authorisation (during 5GS network registration, data session establishment or UAV-GCS pairing), UAV flight authorisation, UAV and GCS authorisation, procedures for re-authentication/re-authorisation and revocation on the UTM's request, location reporting of individual UAV or a list of UAV in a specified area, establishment and control of requested QoS/traffic filtering for C2 communication, or UAV presence monitoring. UAS NF can also implement and expose the CAPIF-compliant interfaces to the UTM ecosystem, as long as CAPIF is deployed in the network.
- NTN systems—3GPP works intensively on integration of NTN components—Geostationary Earth Orbit (GEO), Low Earth Orbit (LEO), and Medium Earth Orbit (MEO) satellites, High Altitude Platforms (HAPs)—into 5GS to accommodate most pressing NTN applications identified so far: coverage extension, IoT, disaster communication, global roaming, and broadcasting [49]. The consolidated functional and QoS-related satellite service requirements have been established (e.g., delay,

expected data rates, traffic capacities) with the most elevated ones for airplane connectivity (360 Mbps downlink, 180 Mbps uplink, planes flying at maximum speed of 1000 km/h) [50]. The scope of NTN RAN is so far limited to a bent-pipe or relay communication mode [51]. Currently, 5GS provides support for NTN in terms of recognition of NTN access by UE, identification/restriction of using satellite 5G radio access, integrated access and backhaul, discontinuous network coverage for satellite access, high latency communication, UPF deployment on satellite (enabling edge computing and local UE–UE communications). Also, the guidelines for extraterritorial 5GSs have been established [52]. The future work proposed for the scope of the 3GPP Release 19 [53] will be focused on 5GS support for new use cases such as store and forward for a delay-tolerant/non-RT IoT in remote areas, ephemeral satellite access-based emergency reporting, UAVs connectivity over satellite access, video streaming, exploiting satellite connectivity to collect supplementary information, i.a., to aid service differentiation for UEs via satellite access.

Accommodation of the advanced 5GS-based services requires the development of mechanisms that boost the performance and capabilities of RAN as well as relevant investments in the infrastructure during the network planning and implementation phases. The key operational aspects related to 5GS RAN are described below.

- Operating frequencies and capacity—5GS is designed to operate in two frequency bands referred to as FR1 (410–7125 MHz) and FR2 consisting of two sub-bands FR2-1 (24,250–52,600 MHz) and FR2-2 (52,600–71,000 MHz) [54,55]. While FR1 and FR2-1 bands are licensed, the FR2-2 is available for unlicensed operation, depending on the regional and/or country-specific regulatory requirements. The available channel widths are: in FR1 (sub-GHz) 5–35 MHz (5 MHz steps), in FR1 (>1 GHz) 5/10/20–100 MHz (depending on a specific band, n $\times$ 5 MHz steps), in FR2-1 50/100/200/400 MHz, and in FR2-2 100/400/800/1600/2000 MHz. For the assumed peak spectrum efficiency of 30 bit/s/Hz [38], the indicative maximum total capacity is 30 Mbit/s per every 1 MHz of used bandwidth. Additionally, bands aggregation (so-called "carrier aggregation") is possible, up to 6 bands in FR1 and up to 2 bands in FR2.
- Spectrum properties and outcomes—5GS operates in a very wide spectrum of frequencies resulting in the occurrence of different radio propagation-related phenomena and their varying intensity, which affect the transmission range. The primary effects that impact the link budget include path loss (i.e., the loss of power due to radio signal propagation, increasing with frequency and distance), scattering (e.g., due to atmospheric conditions like rain or mist), absorption, etc. While the 5GS FR1 frequencies enable maintenance of considerable coverage ranges (over km), the wide-scale exploitation of FR2 poses significant challenges, mostly due to tremendous signal losses (e.g., non-VLOS urban scenario indicated the effective range of approximately 50 m to maintain the throughput of 1 Gb/s for transmission at 27 GHz) [56]. As a consequence, each of the above-mentioned 5GS frequency ranges will play a different role in an overall network picture, i.e., sub-GHz FR1 is expected to cover large areas with relatively low capacity (scarcely populated rural areas), the rest of FR1 band is destined to provide high-capacity city-wide coverage, while the highest FR2-1, FR2-2 will deliver ultra-high gigabit speeds and low latencies on very short distances (applicable mostly in campus network deployments) [57].
- Specific support and RAN-related enablers—The already described 5GS NS is an E2E concept, i.e., it applies both to CN and RAN. While the slicing in CN is well defined, the latter is left to the equipment vendors, which opens the way for the inclusion of dedicated support for specific use cases, if needed. RAN slicing is usually achieved by implementing traffic handling policies and relevant prioritisation (e.g., special treatment of AAM traffic) in the base station at the scheduler level to meet QoS targets for each slice. It must be noted that 3GPP also does not specify the architecture of the RAN controllers. This approach allows for using third-party solutions such as Open RAN (O-RAN), which provides the support for generic RAN optimising applications

(so-called xAPPs) that can be tailored to the AAM benefits and specific needs [58]. Moreover, 3GPP proposes several RAN mechanisms to address the elevated performance requirements. One of the common concerns regarding 5GS-based AAM is the aspect of coverage provisioning. To this end, 5GS provides functionalities of beam management (beam forming, steering, selection) [51] that enable the improvement of the quality of transmission between UE and the base station. For today, however, the means for leveraging this mechanism in cooperation with UTM is very limited (no interfaces to request coverage adaption in the area of concern directly is defined).

Concluding, 5GS is a programmable, scalable mobile telecommunications framework, which offers not only reliable communication capabilities but also extensive support to leverage vertical services. The most significant features include CN functions and data exposure via standardised APIs to the vertical ecosystem, implementation of dedicated CN functions, integration with the UTM ecosystem (via dedicated UAS NF) or edge environments, ability to accelerate service deployment by reusing shared generic vertical CN functions (via SEAL framework) and the mechanisms for fine-grained control over the end users' traffic.

5. Evolution towards 6G and Perspectives for AAM

With the high maturity of the 3GPP standardisation and rapidly progressing deployment of 5GS across the world, the work on defining the targets for the next mobile telecommunications system has started. It is commonly agreed that 6GS should be oriented towards the societal and environmental challenges, e.g., stated within the United Nations (UN) Sustainable Development Goals (SDG), emerging market expectations, and operational necessities of MNOs [59]. While discussions on the 6G visions and the applied research on enablers and future technologies supported by 6GS have already begun, the beginning of the international standardisation, and thus formulation of architectural solutions, is expected around 2025, and the first implementations since 2030 [60]. That is why a good understanding of the fundamental trends driving 6G visions by the AAM area, the future customer and user of the new communications technology, is so important at this stage when it is possible to influence directions. This section will present the major 6G drivers, design principles and envisioned capabilities that can contribute to accelerating the development of carrier-grade AAM services.

New applications—6G mobile technology is envisioned around new innovative and disruptive applications that include [61–63]:

- Holographic communications—based on laser beam producing 3-Dimensional (3D) RT motion images (holography) to be used, i.a., for immersive telepresence and 3D FPV. Enabling full holographic support requires data rates in the order of 1–5 Tbps, E2E latency less than 1 ms, and transmission jitter less than a microsecond to provide concurrent multiple viewing angles' streams timing synchronisation, and extreme reliability in most challenging use cases (e.g., remote surgery).
- Multisensory xR—a class of applications that includes AR, VR, and Mixed Reality (MR), i.e., merging physical and computer-generated environments, where objects of both can co-exist and interact in real time. The required data rates are in the order of multiple Gbps up to 1 Tbps, and the E2E latency <15 ms.
- Tactile Internet—haptic interaction of humans or machines with machines used in the manufacturing industry, for telepresence and teleoperation, healthcare (especially telesurgery), an extension of VR/AR. Fundamental requirements here encompass sub-millisecond latency and high security.
- Brain-Computer Interaction (BCI)—enabling interaction of humans with their environment and other people using discrete devices (worn, implanted, or in ambience). High data rates and reliability with ultra-low latency are required. Similarly to the two former ones, the requirements have a stringent perceptual aspect stemming from human neurophysiology of senses and cognition; much more sensitive and thus demanding, however.

- Digital twin—the concept of virtual digital replicas of physical objects, systems, or environments, enabling the simulation of phenomena to be used in research, behavioural prediction, education, medical practice, etc. While it will exist in a computation environment, it can use holographic communications, multisensory xR or haptic mechanisms to interact with the human user, inheriting thus their service requirements.
- Teleoperated driving and unmanned mobility—human control of remote vehicles, also as an option for semi-autonomous vehicles or backup for autonomous vehicles when the autonomous mode fails, or immediate response to danger is necessary. It can be used for land or airborne use cases (especially in far or dangerous areas), but also in deep-sea operations or space exploration. The requirements feature extremely high data rates, low latencies, and high levels of communication reliability, integrity, privacy, and security.
- Connected robotics and autonomous systems—they include autonomous cars and ground vehicles, vehicle platoons, drone-delivery systems, autonomous drone swarms, and autonomous robotics. The requirements include the E2E reliability above 99.99999% and sub-millisecond latency at mobility up to 1000 km/h.

The listed applications, beyond the 5GS capabilities, either have a direct association with the AAM needs and context or may be creatively adapted and reused to address them.

Paradigm shift in the network design—Instead of hitherto "network centricity", "user centricity" [62,64] and "service awareness" are envisioned. The former means a reversal of the relationship—so far, the user has been dependent on the network—its coverage related to network cellularity, capacity, service capabilities, etc. In the case of 6G, it can be said that the network attaches to the user, following him and actively adapting to the service needs to deliver fully and individually user-definable, user-configurable, and user-controllable communication services, e.g., by optimising the parameters of an individual radio link from one or multiple base stations (a cell-free concept with dynamic and overlapping base stations topology) to ensure the appropriate quality of connection in the UE location or by dynamically installing the service chain for the user. The latter is associated with a significantly expanded set of communication services with distinct characteristics compared to 5G, where only three basic classes defined by ITU [37] were recognised: eMBB, Ultra-Reliable Low-Latency Communication (URLLC) and Massive Machine Type Communications (mMTC). Depending on the vision [61,62,65,66] with different naming used, numerous classes of services are defined, but the basic idea behind them is to create diverse blends of the above triad, taking into account various goals for such parameters as peak/perceived data rate, latency including its jitter, reliability, traffic capacity, connection density, user mobility (UE motion velocity), positioning accuracy, etc.

Convergence of multiple functional dimensions—Unlike the earlier generations that are purely communication-oriented, multifaceted convergence is proposed depending on the vision of 6GS—Communication, Computing and Caching (3C) [67], Communication, Computation, Caching and Control (C4), and Communications, Computing, Control, Localisation and Sensing (3CLS) [61]. It will define 6GS as a convergent, multi-purpose system able to deliver various services and applications hosted therein based on these functions as inherent and native features. Another related concept from this area is Integrated Sensing and Communication (ISAC) [68,69], which offers Remote Sensing and Sensing as a Service (SaaS) that also cover the localisation aspect. Examples of ISAC applications include drone swarm Synthetic-Aperture Radar (SAR) imaging, drones monitoring and management (including drones recognition and tracking), enhanced localisation and tracking in mobile network, spatial-aware computing (exploiting geometric relationships between devices), and simultaneous localisation and mapping for building situational awareness from environment exploring, imaging and reconstruction. The convergent functions will be placed in all 6GS points of presence, including the non-terrestrial ones, to enable shifting the user application processing as close to the user as possible.

Pervasive AI—While AI in 5GS is an optional add-on, a collective, distributed AI will be an inherent, endogenous component of 6GS to enable autonomy of network operations, better performance, and resources optimisation, in particular in the evolution from a self-organising network (4G–5G) to self-sustaining, intelligent network, e.g., employing Zero-touch network and Service Management (ZSM) [70] or Experiential Networked Intelligence (ENI) [71] mechanisms to provide autonomous network (including dynamic E2E network slicing) and service orchestration and management [72]. AI will also support the implementation of the "cognitive RAN" concept, i.e., dynamic and intelligent sensing of the electromagnetic environment by base stations in order to find spectrum available for local use. AI-driven user context awareness recognition and analysis, per-user predictive analysis of service degradation will support the user-centricity aspect of 6GS, QoS improvement, and fulfilment of SLA. Additionally, through the Artificial Intelligence as a Service (AIaaS) approach and leveraging the convergence described above, the network intelligence can be consumed by user services. This relation, however, will be mutual: due to the 3C/C4/3CLS, the intelligence will migrate from the central cloud to far edges-located ubiquity and closeness to the user.

Ubiquitous 3D connectivity—Unlike the previous generations, 6GS will be, by design, a 3D omnipresent platform with 4 tiers: terrestrial, aerial, space, and underwater [73] to support any communications demand anywhere. From the point of view of AAM, the aerial and space tiers will be of premium importance in terms of providing communications services availability in current white spots with reasonable delays related to altitudes; the former implemented with floating base stations in HAPs and flying base stations in UAVs (altitude range 8–50 km and footprint range 5–200 km [74], round trip delay component due to a distance of less than 2 ms), the latter—when utilising constellations of LEO satellites (altitude range 300–1500 km and footprint range 100–1000 km [74], round trip delay component due to distance of less than 25 ms). It should be noted that LEO satellites orbit with tangential velocities of 7.5 km/s (27,000 km/h) and orbital periods of 1.5–2 h [75], it is, therefore, necessary to develop new modulation schemes resistant to Doppler shift and robust UE mobility management mechanisms. Applicability of unified multi-tier UEs in AAM vehicles will also require reducing the dimensions of antennas for communication with LEO satellites without degradation of the link signal budget, e.g., through distributed beamforming from a large virtual array composed of LEO satellites swarms [76].

Improved service availability and TTM—6GS is expected to support cloud-continuum concept [77], i.e., a highly heterogeneous infrastructure layer with resources managed by multiple owners, via tailored management and orchestration systems [78]. The ability to exploit the wide availability of resources will allow for a massive improvement of spatial service availability via dynamic deployment or migration of NF (especially important in the context of UAVs operations and U-space services) as well as optimisation via optimal placement strategies of softwarised NFs.

6. Evolution of Mobile Systems Support of AAM Requirements

The feasibility of AAM applications in mobile networks depends not only on the functional properties and architectures of these networks but also on network performance, which translates into deliverable QoS. In Table 3, the comparison of key 5G and future 6G network performance requirements is provided. Compared to 5GS, most performance requirements of 6GS are expected to improve by even several orders of magnitude. Due to the assumed 3D ubiquity, indicators related to service capacity and demand density are changing from 2-Dimensional (2D) to volumetric ones. The increase in the UE mobility support index enables the provision of services for passenger aviation. There is also a definition of maximum delay jitter, which will enable support for deterministic networking, in particular, holographic communication. However, it should be emphasised that the existing system with defined architecture and properties, subject to the systematic development of standardisation, is compared here with visions of the new system, which is based on postulates and expectations.

Table 3. Comparison of 5G and 6G target KPIs.

Network Performance Requirement	5G KPI	6G KPI
Peak downlink data rate [64]	20 Gbps	1–10 Tbps
Peak uplink data rate [64]	10 Gbps	1–10 Tbps
Experienced downlink data rate [64]	100 Mbps	10–100 Gbps
Experienced uplink data rate [64]	50 Mbps	10–100 Gbps
CP delay [64]	10 ms	1 ms
UP delay [64,79]	1 ms	0.1 ms
Delay jitter [79]	–	1 µs
Connection density [64]	10^6 dev./km^2	10^7–10^8 dev./km^3
Traffic capacity [64]	10 Mbps/m^2	0.1–10 Gbps/m^2
Reliability [64]	99.999%	99.99999%
Spectral efficiency [80]	30 bps/Hz	100 bps/Hz
Mobility support [80]	500 km/h	1000 km/s
Positioning accuracy [64]	meter-level	sub-meter level (outdoor), centimeter-level (indoor)

From the point of view of QoS requirements, it should be checked whether the most demanding QoS targets of potential AAM applications presented in Sections 2, 3 and 5 can be met by PLMN. The specificity of AAM applications should be taken into account, in which case there is no domination of the downlink traffic, and sometimes the uplink traffic may even be dominant. Based on [11,50,81,82], the following KPIs can be validated:

- Data rate: 1–5 Tbps (holographic communication, 6G), <1 Tbps (multisensory xR, 6G), several Gbps (uplink, near-RT photogrammetry), 1.1 Gbps (remote robots), 0.1–1 Gbps (AR/VR), 10 s–100 s Mbps (downlink, on-board broadband connectivity sharing), 120 Mbps (uplink, RT multimedia video surveillance, very high definition FPV for remote pilot), 3.2 Mbps (the most demanding C2 mode—approaching autonomous navigation infrastructure);
- Latency (delay): <1 ms (holographic communication, tactile Internet, connected robotics and autonomous systems, all 6G), 2 ms (remote robots), 10 ms (AR/VR, the most demanding C2 mode—approaching autonomous navigation infrastructure), 20 ms (RT multimedia video surveillance and FPV for remote pilot), 40 ms (C2 for manual steering), 500 ms (connectivity to UTM);
- Delay jitter: <1 µs (holographic communication, 6G);
- Reliability: 99.99999% (connected robotics and autonomous systems, 6G), 99.99% (RT multimedia video surveillance and FPV for remote UAV pilot, AR/VR), 99.9% (remote robots, most C2 modes, connectivity to UTM); note: according to 3GPP, reliability is "in the context of network layer packet transmissions, the percentage value of the packets successfully delivered to a given system entity within the time constraint required by the targeted service out of all the packets transmitted" [50].

The analysis of the above requirements in the context of Table 3 shows that even some of the non-6G AAM applications will not be able to be satisfyingly implemented in 5GS, primarily due to the data rate (near-RT photogrammetry, on-board broadband connectivity sharing) and latency (in practice, 2 ms E2E latency will only be possible in campus networks). While the peak traffic capacity of 5GS reaches 10/20 Gbps, it should be remembered that the values achieved by users strongly depend on the spatial structure of service demand, i.e., in the first approximation, the number of UEs served by a given base station.

One of the key issues to be solved by 6GS is the aspect of real network ubiquity, not only 3D one, primarily the elimination of 2D white spots. Despite the widespread deployment of 5GS worldwide, access to 5G network services is still limited to highly urbanised areas. In the case of EU, the estimations show that only around 72% of EU population can use the 5G network services [83], which indicates the massive deficiency of coverage in rural areas. Additionally, the 5G networks are mainly designed as metropolitan networks, as the 3GPP definition for rural macro scenarios [50] assumes the user-experienced data rate of 50 Mbps (downlink) and 25 Mbps (uplink), while traffic capacities are 1 Gbps/km^2 (downlink) and 0.5 Gbps/km^2 (uplink). For comparison, in the urban macro scenario, 200× higher data rates and 100× better traffic capacities are assumed for both uplink and downlink. Therefore, in non-urban 5G points-of-presence, despite the expected lower number of attached UEs, the experienced data rate will be lower than in urban areas.

Systematic functional (standardisation) and coverage (MNOs' investments) development of the 5G network may support most AAM applications, mainly in urban areas. It should be expected that the gradual migration to 6G after 2030 will improve coverage and increase the perceived data rate, as well as lower the perceived latency for the applications listed in Section 2. This will be conditioned primarily by ensuring similar traffic capacity of aerial and space (LEO) tiers base stations, which will play a major role in rural and uninhabited areas. Based on the experience gained so far (in particular the evolution of the 5GS standardisation), it can be assumed that support for the applications discussed in Section 5 outside metropolitan areas will appear later, however, the implementation of 6GS will open new functional horizons that will support the development of AAM.

7. Summary and Conclusions

In this paper, the mobile network evolution in the context of AAM has been presented. First, the high-level picture of AAM ecosystem has been drawn together with conditions specifics of the aviation domain. New emerging use cases within four AAM use case groups, namely: human mobility, goods mobility, data collection and task/process automation, have been identified and described. Also, the benefits of multi-purpose AAM missions, i.e., performing in-flight tasks, additional to the primary mission goal, such as, e.g., environmental and network data collection, extending network coverage, photogrammetry, etc. Afterwards, an extensive review of 3GPP standardisation related to AAM-specific support has been outlined together with the most notable research achievements concerning the integration of aviation and mobile ecosystems. The high maturity of standards and solutions on both aviation and 3GPP sides provides the solid foundation for the future development of AAM services. Moreover, the key 5GS paradigms, architectural solutions, support for AAM verticals (network capability exposure, operation in edge environments, integration with UTM or NTN systems), and RAN-related features that can positively contribute to AAM operations have been described in detail. It must be emphasised that for today, whereas 5GS provides solid support for UAV services, some major issues still exist, e.g., coverage and service availability in remote rural areas. Nonetheless, 3GPP progresses in this matter, e.g., by advancing the integration of 5GS with NTNs such as satellite systems or HAPs. Finally, the envisioned evolution towards 6G and resulting future perspectives for AAM are discussed. While the visions of 6GS have stabilised, and there exists a consensus among the research community regarding the required capabilities and characteristics of the system, the official standardisation activities have not started, yet. Nonetheless, it can already be stated that 6GS will be a user-centric system focused on providing quality on a per-user basis. To this end, it is envisioned that 6GS will provide ubiquitous 3D coverage by leveraging the integration of multiple communication networks (i.e., implementing the "network of networks" paradigm), leverage distributed AI-based operations, integrate AI in the system design, feature high modularity and flexibility, that all together will allow for the accommodation of innovative and disruptive applications. Whereas the requirements of the most of considered AAM use cases are satisfied by 5GS, to host more demanding applications, significant improvements regarding the offered data rate (e.g., for RT photogrammetry) and E2E latency are needed. To this end, from the AAM perspective, 6GS will not only provide means for the implementation of more challenging use cases but will also contribute to the reduction of TTM for the AAM verticals, collectively leading to the solid acceleration of the AAM market growth.

Author Contributions: Conceptualisation L.T. and R.K.; methodology, L.T. and R.K.; visualisation, L.T. and R.K.; writing—original draft, L.T. and R.K.; writing—review and editing L.T. and R.K. All authors have read and agreed to the published version of the manuscript.

Funding: ETHER (https://doi.org/10.3030/101096526) project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101096526. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

2D	2-Dimensional
3C	Communication, Computing and Caching
3CLS	Communications, Computing, Control, Localisation and Sensing
3D	3-Dimensional
3GPP	3rd Generation Partnership Project
5G-EIR	5G Equipment Identity Register
5GS	5G System
6GS	6G System
AAM	Advanced Air Mobility
ACJA	Aerial Connectivity Joint Activity
ADRF	Analytics Data Repository Function
AF	Application Function
AI	Artificial Intelligence
AIaaS	Artificial Intelligence as a Service
AMF	Access and Mobility Management Function
API	Application Programming Interface
AR	Augmented Reality
ATM	Air Traffic Management
AUSF	Authentication Server Function
BCI	Brain-Computer Interaction
BVLOS	Beyond Visual Line of Sight
C2	Command and Control
C4	Communication, Computation, Caching and Control
CAPIF	Common API Framework for 3GPP northbound APIs
CN	Core Network
CP	Control Plane
CUPS	Control and User Plane Separation
DAA	Detect And Avoid
DCCF	Data Collection Coordination Function
DN	Data Network
E2E	end-to-end
EASA	European Union Aviation Safety Agency
eMBB	Enhanced Mobile Broadband
ENI	Experiential Networked Intelligence
ETSI	European Telecommunications Standards Institute
EU	the European Union
FPV	First Person View
GCS	Ground Control Station
GEO	Geostationary Earth Orbit

GSMA	GSM Association
GUTMA	Global UTM Association
HAP	High Altitude Platform
HW	hardware
ICT	Information and Communications Technology
IoT	Internet of Things
IP	Internet Protocol
ISAC	Integrated Sensing and Communication
ISM	Industrial, Scientific, Medical
ITU	International Telecommunication Union
KPI	Key Performance Indicator
LCS	LoCation Services
LEO	Low Earth Orbit
LoRaWAN	ong Range Wide Area Network
MANO	Management and Orchestration
MEC	Multi-access Edge Computing
MEO	Medium Earth Orbit
MFAF	Messaging Framework Adaptor Function
mMTC	Massive Machine Type Communications
MNO	Mobile Network Operator
MR	Mixed Reality
NASA	National Aeronautics and Space Administration
NEF	Network Exposure Function
NF	Network Function
NRF	Network Repository Function
NS	Network Slicing
NSACF	Network Slice Admission Control Function
NSSAAF	Network Slice-specific and Stand-alone non-public network Authentication and Authorisation Function
NSSF	Network Slice Selection Function
NTN	Non-Terrestrial Network
NWDAF	NetWork Data Analytics Function
O-RAN	Open RAN
PCF	Policy Control Function
PLMN	Public Land Mobile Network
QoS	Quality of Service
RAN	Radio Access Network
RT	Real-Time
SaaS	Sensing as a Service
SAR	Synthetic-Aperture Radar
SBA	Service-Based Architecture
SCP	Service Communication Proxy
SDG	Sustainable Development Goals
SEAL	Service Enabler Architecture Layer
SESAR	Single European Sky ATM Research
SLA	Service Level Agreement
SMF	Session Management Function
SMSF	Short Message Service Function
SW	software
TN	Terrestrial Network
TPAE	Third-Party Authorised Entity
TTM	Time To Market
UAM	Urban Air Mobility
UAS	Unmanned Aviation System
UAS NF	Unmanned Aviation System Network Function
UAV	Unmanned Aerial Vehicle
UDM	Unified Data Management
UDR	Unified Data Repository

UDSF	Unstructured Data Storage Function
UE	User Equipment
UN	United Nations
UP	User Plane
UPF	User Plane Function
URLLC	Ultra-Reliable Low-Latency Communication
UTM	Unmanned Aircraft Systems Traffic Management
VLOS	Visual Line of Sight
VR	Virtual Reality
VTOL	Vertical Take-Off and Landing
ZSM	Zero-touch network and Service Management

References

1. NASA. Advanced Air Mobility Project. Available online: https://www.nasa.gov/aeroresearch/programs/iasp/aam/description/ (accessed on 29 May 2023).
2. National Academies of Sciences, Engineering, and Medicine. *Advancing Aerial Mobility: A National Blueprint*; The National Academies Press: Washington, DC, USA, 2020. [CrossRef]
3. EASA. Urban Air Mobility (UAM). Available online: https://www.easa.europa.eu/en/domains/urban-air-mobility-uam (accessed on 29 May 2023).
4. GUTMA. *UAS Traffic Management Architecture*; Version 1.0; Global UTM Association: Lausanne, Switzerland, 2017. Available online: https://www.gutma.org/docs/Global_UTM_Architecture_V1.pdf (accessed on 29 May 2023).
5. Concept of Operations for EUROPEAN UTM Systems—Extension for Urban Air Mobility (CORUS-XUAM Project). U-space ConOps (edition 3.10). 2022. Available online: https://corus-xuam.eu/wp-content/uploads/2022/11/CORUS-XUAM-D4.1-delivered_3.10.pdf (accessed on 29 May 2023).
6. *Technical Standard TS 23.501*; System Architecture for the 5G System (5GS). Version 17.9.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3144 (accessed on 29 May 2023).
7. Ericsson. *5G Advanced: Evolution towards 6G*; White Paper Bnew-22:024836 Uen; Ericsson: Stockholm, Sweden, 2023. Available online: https://www.ericsson.com/49ce19/assets/local/reports-papers/white-papers/5g-advanced-evolution-towards-6g-v2.pdf (accessed on 29 May 2023).
8. 3GPP. 3GPP Releases. Available online: https://portal.3gpp.org/#/55934-releases (accessed on 29 May 2023).
9. Huawei. 6G: The Next Horizon from Connected People and Things to Connected Intelligence. White Paper, Huawei: Shenzen, China, 2021. Available online: https://www-file.huawei.com/-/media/corp2020/pdf/tech-insights/1/6g-white-paper-en.pdf?la=en (accessed on 29 May 2023).
10. 3GPP. *Study on Remote Identification of Unmanned Aerial Systems (UAS)*; Technical Report TR 22.825; Version 16.0.0; 3rd Generation Partnership Project: Valbonne, France, 2018. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3527 (accessed on 29 May 2023).
11. *Technical Standard TS 22.125*; Unmanned Aerial System (UAS) Support in 3GPP. Version 17.6.0; 3rd Generation Partnership Project: Valbonne, France, 2022. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3545 (accessed on 29 May 2023).
12. 3GPP. *Enhancement for Unmanned Aerial Vehicles (UAVs)*; Technical Report TR 22.829; Version 17.1.0, 3rd Generation Partnership Project: Valbonne, France, 2019. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3557 (accessed on 29 May 2023).
13. 3GPP. *Study on Uncrewed Aerial Vehicle (UAV) Phase 3 (Release 19)*; Technical Report TR 22.843; Version 19.0.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4090 (accessed on 29 May 2023).
14. 3GPP. *Study on Supporting Unmanned Aerial Systems (UAS) Connectivity, Identification and Tracking*; Technical Report TR 23.754; Version 17.1.0; 3rd Generation Partnership Project: Valbonne, France, 2021. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3575 (accessed on 29 May 2023).
15. 3GPP. *Study on Application Layer Support for Unmanned Aerial Systems (UAS)*; Technical Report TR 23.755; Version 17.0.0; 3rd Generation Partnership Project: Valbonne, France, 2021. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3588 (accessed on 29 May 2023).
16. *Technical Standard TS 23.256*; Support of Uncrewed Aerial Systems (UAS) Connectivity, Identification and Tracking; Stage 2. Version 18.1.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3853 (accessed on 29 May 2023).
17. *Technical Standard TS 23.255*; Application Layer Support for Uncrewed Aerial System (UAS); Functional Architecture and Information Flows. Version 18.2.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3843 (accessed on 29 May 2023).

18. 3GPP. *Study of Further Architecture Enhancements for Uncrewed Aerial Systems and Urban Air Mobility*; Technical Report TR 23.700-58; Version 18.1.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3993 (accessed on 29 May 2023).
19. 3GPP. *Study on Enhanced Application Architecture for UAS Applications*; Technical Report TR 23.700-55; Version 18.0.0; 3rd Generation Partnership Project: Valbonne, France, 2022. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3945 (accessed on 29 May 2023).
20. 3GPP. *Study on Security Aspects of Uncrewed Aerial Systems (UAS)*; Technical Report TR 33.854; Version 17.1.0; 3rd Generation Partnership Project: Valbonne, France, 2021. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3763 (accessed on 29 May 2023).
21. *Technical Standard TS 33.256*; Security Aspects of Uncrewed Aerial Systems (UAS). *Version 17.3.0*; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3930 (accessed on 29 May 2023).
22. 3GPP. *Study on Security of Architecture Enhancements for UAS, UAV and UAM*; Technical Report TR 33.891; Version 18.0.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4083 (accessed on 29 May 2023).
23. *Technical Standard TS 29.256*; Uncrewed Aerial Systems Network Function (UAS-NF); Aerial Management Services; Stage 3. Ver. 18.1.0, 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3866 (accessed on 29 May 2023).
24. *Technical Standard TS 29.255*; Uncrewed Aerial System Service Supplier (USS) Services; Stage 3. Version 18.1.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3904 (accessed on 29 May 2023).
25. 3GPP. *Release 17 Description; Summary of Rel-17 Work Items*; Technical Report TR 21.917; Version 17.0.1; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3937 (accessed on 29 May 2023).
26. 3GPP. NR Support for UAVs. Available online: https://www.3gpp.org/technologies/nr-uav/ (accessed on 29 May 2023).
27. 5G!Drones: Unmanned Aerial Vehicle Vertical Applications' Trials Leveraging Advanced 5G Facilities. Available online: https://5gdrones.eu (accessed on 29 May 2023).
28. Tomaszewski, L.; Kołakowski, R.; Kukliński, S. Integration of U-space and 5GS for UAV services. In Proceedings of the 2020 IFIP Networking Conference (Networking), Paris, France, 22–25 June 2020; pp. 767–772. Available online: https://ieeexplore.ieee.org/document/9142706 (accessed on 29 May 2023).
29. Tomaszewski, L.; Kołakowski, R.; Korzec, P. On 5G Support of Cross-Border UAV Operations. In Proceedings of the 2020 IEEE International Conference on Communications Workshops (ICC Workshops), Dublin, Ireland, 7–11 June 2020; pp. 1–6. [CrossRef]
30. PriMO-5G: Virtual Presence in Moving Objects through 5G. Available online: https://primo-5g.eu/ (accessed on 29 May 2023).
31. GOF 2.0 Integrated Urban Airspace Validation. Available online: https://gof2.eu/project (accessed on 29 May 2023).
32. GOF 2.0 Consortium. Validation of Integrated Multi-stakeholder Architecture for UAS Traffic Management: GOF 2.0 Integrated Urban Airspace. White Paper, 2022. Available online: https://gof2.eu/wp-content/uploads/2022/12/GOF2.0-white-paper.pdf (accessed on 29 May 2023).
33. MITRE Engenuity. Open Generation Uncrewed Aircraft Systems (UAS) Use Cases. Report, Next Generation Mobile Networks, 2022. Available online: https://f.hubspotusercontent20.net/hubfs/7754670/Open_Generation/Open%20Gen%20Reports/ME_OpenGen_UAS_UseCases_Feb2022.pdf (accessed on 29 May 2023).
34. ACJA. Aerial Connectivity Joint Activity (ACJA) Resources. Available online: https://www.gsma.com/iot/advanced-air-mobility-resources/ (accessed on 29 May 2023).
35. Tomaszewski, L.; Kołakowski, R. Network Slicing vs. Network Neutrality—Is Consent Possible? In Proceedings of the Artificial Intelligence Applications and Innovations; AIAI 2023 IFIP WG 12.5 International Workshops, León, Spain, 14–17 June 2023; Maglogiannis, I., Iliadis, L., Papaleonidas, A., Chochliouros, I., Eds.; Springer Nature Switzerland: Cham, Switzerland, 2023; pp. 77–90. [CrossRef]
36. ETSI. *Network Functions Virtualisation (NFV); Management and Orchestration*; Group Specification ETSI GS NFV-MAN 001 V1.1.1; European Telecommunications Standards Institute: Valbonne, France, 2014. Available online: https://www.etsi.org/deliver/etsi_gs/nfv/001_099/001/01.01.01_60/gs_nfv001v010101p.pdf (accessed on 29 May 2023).
37. ITU-R. *IMT Vision–Framework and Overall Objectives of the Future Development of IMT for 2020 and Beyond*; Recommendation ITU-R M.2083-0 (09/15); International Telecommunication Union—Radiocommunication Sector: Geneva, Switzerland, 2015. Available online: https://www.itu.int/rec/R-REC-M.2083-0-201509-I/ (accessed on 29 May 2023).
38. ITU-R. *Minimum Requirements Related to Technical Performance for IMT-2020 Radio Interface(s)*; Report ITU-R M.2410-0 (11/17); International Telecommunication Union—Radiocommunication Sector: Geneva, Switzerland, 2017. Available online: https://www.itu.int/pub/R-REP-M.2410-2017 (accessed on 29 May 2023).
39. *Technical Standard TS 23.501*; System Architecture for the 5G System (5GS). Version 18.2.2; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3144 (accessed on 29 May 2023).

40. *Technical Standard TS 23.273*; 5G System (5GS) Location Services (LCS); Stage 2. Version 18.2.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3577 (accessed on 29 May 2023).
41. *Technical Standard TS 23.548*; 5G System Enhancements for Edge Computing; Stage 2. Version 18.2.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3856 (accessed on 29 May 2023).
42. *Technical Standard TS 23.222*; Common API Framework for 3GPP Northbound APIs. Version 18.2.0, 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3337 (accessed on 29 May 2023).
43. *Technical Standard TS 23.434*; Service Enabler Architecture Layer for Verticals (SEAL); Functional Architecture and Information Flows. Version 18.5.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3587 (accessed on 29 May 2023).
44. *Technical Standard TS 23.558*; Architecture for Enabling Edge Applications. Version 18.3.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3723 (accessed on 29 May 2023).
45. ETSI. *Multi-access Edge Computing (MEC); Radio Network Information API*; Group Specification ETSI GS MEC 012 V2.2.1; European Telecommunications Standards Institute: Valbonne, France, 2022. Available online: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/012/02.02.01_60/gs_mec012v020201p.pdf (accessed on 29 May 2023).
46. ETSI. *Multi-Access Edge Computing (MEC); Location API*; Group Specification ETSI GS MEC 013 V3.1.1; European Telecommunications Standards Institute: Valbonne, France, 2023. Available online: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/013/03.01.01_60/gs_mec013v030101p.pdf (accessed on 29 May 2023).
47. ETSI. *Multi-Access Edge Computing (MEC); UE Identity API*; Group Specification ETSI GS MEC 014 V3.1.1; European Telecommunications Standards Institute: Valbonne, France, 2023. Available online: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/014/03.01.01_60/gs_mec014v030101p.pdf (accessed on 29 May 2023).
48. ETSI. *Multi-Access Edge Computing (MEC); Traffic Management APIs*; Group Specification ETSI GS MEC 015 V2.2.1; European Telecommunications Standards Institute: Valbonne, France, 2022. Available online: https://www.etsi.org/deliver/etsi_gs/MEC/001_099/015/02.02.01_60/gs_mec015v020201p.pdf (accessed on 29 May 2023).
49. 3GPP. *Study on Using Satellite Access in 5G*; Technical Report TR 22.822; Version 16.0.0; 3rd Generation Partnership Project: Valbonne, France, 2018. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3372 (accessed on 29 May 2023).
50. *Technical Standard TS 22.261*; Service Requirements for the 5G System; Stage 1; Version 19.3.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3107 (accessed on 29 May 2023).
51. *Technical Standard TS 38.300*; NR; NR and NG-RAN Overall Description; Stage-2; Version 17.5.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3191 (accessed on 29 May 2023).
52. 3GPP. *Guidelines for Extraterritorial 5G Systems (5GS)*; Technical Report TR 22.926; Version 18.0.0; 3rd Generation Partnership Project: Valbonne, France, 2021. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3722 (accessed on 29 May 2023).
53. 3GPP. *Study on Satellite Access—Phase 3*; Technical Report TR 22.865; Version 19.0.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4089 (accessed on 29 May 2023).
54. *Technical Standard TS 38.101-1*; NR; User Equipment (UE) Radio Transmission and Reception; Part 1: Range 1 Standalone. Version 18.2.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3283 (accessed on 29 May 2023).
55. 3GPP. *Technical Standard TS 38.101-2*; NR; User Equipment (UE) Radio Transmission and Reception; Part 2: Range 2 Standalone. Version 18.2.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3284 (accessed on 29 May 2023).
56. ITU-R. Technical Feasibility of IMT in Bands above 6 GHz. Recommendation ITU-R M.2376-0 (07/2015), International Telecommunication Union—Radiocommunication Sector, 2015. Available online: https://www.itu.int/pub/R-REP-M.2376-2015 (accessed on 29 May 2023).
57. GSMA. *5G Spectrum: GSMA Public Policy Position*; Position Report; GSM Association: London, UK, 2022. Available online: https://www.gsma.com/spectrum/wp-content/uploads/2022/06/5G-Spectrum-Positions.pdf (accessed on 29 May 2023).
58. O-RAN Alliance. O-RAN Use Cases and Deployment Scenarios. White Paper, Open RAN Alliance, 2020. Available online: https://assets-global.website-files.com/60b4ffd4ca081979751b5ed2/60e5aff9fc5c8d496515d7fe_O-RAN%2BUse%2BCases%2Band%2BDeployment%2BScenarios%2BWhitepaper%2BFebruary%2B2020.pdf (accessed on 29 May 2023).
59. NGMN. NGMN 6G Drivers and Vision. White Paper Version 1.0, Next Generation Mobile Networks, 2021. Available online: https://www.ngmn.org/wp-content/uploads/NGMN-6G-Drivers-and-Vision-V1.0_final_New.pdf (accessed on 29 May 2023).

60. Magedanz, T.; Corici, M.I. Getting ready for 6G Research—Understanding Technological Drivers towards 6G and emerging 6G Management Requirements. Tutorial at IEEE/IFIP NOMS, 25 April 2022, Fraunhofer FOKUS: Berlin, Germany, 2022. Available online: https://owncloud.fokus.fraunhofer.de/index.php/s/4oIylfMlANQTZCB/download (accessed on 29 May 2023).
61. Imoize, A.L.; Adedeji, O.; Tandiya, N.; Shetty, S. 6G Enabled Smart Infrastructure for Sustainable Society: Opportunities, Challenges, and Research Roadmap. *Sensors* **2021**, *21*, 1709. [CrossRef]
62. Saad, W.; Bennis, M.; Chen, M. A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems. *IEEE Netw.* **2020**, *34*, 134–142. [CrossRef]
63. NGMN. 6G Use Cases and Analysis. White Paper Version 1.0, Next Generation Mobile Networks, 2022. Available online: https://www.ngmn.org/wp-content/uploads/220222-NGMN-6G-Use-Cases-and-Analysis-1.pdf (accessed on 29 May 2023).
64. Liu, G.; Huang, Y.; Li, N.; Dong, J.; Jin, J.; Wang, Q.; Li, N. Vision, requirements and network architecture of 6G mobile network beyond 2030. *China Commun.* **2020**, *17*, 92–104. [CrossRef]
65. Mahmood, N.; Böcker, S.; Munari, A.; Clazzer, F.; Moerman, I.; Mikhaylov, K.; Alcaraz López, O.; Park, O.S.; Mercier, E.; Bartz, H.; et al. White Paper on Critical and Massive Machine Type Communication Towards 6G. In *6G Research Visions*; University of Oulu: Oulu, Finland, 2020; Volume 11. Available online: http://urn.fi/urn:isbn:9789526226781 (accessed on 29 May 2023).
66. Alves, H.; Jo, G.D.; Shin, J.; Yeh, C.; Mahmood, N.H.; Lima, C.; Yoon, C.; Rahatheva, N.; Park, O.S.; Kim, S.; et al. Beyond 5G URLLC Evolution: New Service Modes and Practical Considerations. *arXiv* **2021**, arXiv:2106.11825.
67. Zhou, Y.; Liu, L.; Wang, L.; Hui, N.; Cui, X.; Wu, J.; Peng, Y.; Qi, Y.; Xing, C. Service-aware 6G: An intelligent and open network based on the convergence of communication, computing and caching. *Digit. Commun. Netw.* **2020**, *6*, 253–260. [CrossRef]
68. Cui, Y.; Liu, F.; Jing, X.; Mu, J. Integrating Sensing and Communications for Ubiquitous IoT: Applications, Trends, and Challenges. *IEEE Netw.* **2021**, *35*, 158–167. [CrossRef]
69. Liu, F.; Cui, Y.; Masouros, C.; Xu, J.; Han, T.X.; Eldar, Y.C.; Buzzi, S. Integrated Sensing and Communications: Toward Dual-Functional Wireless Networks for 6G and Beyond. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 1728–1767. [CrossRef]
70. ETSI. Zero Touch Network & Service Management (ZSM). Available online: https://www.etsi.org/committee/zsm (accessed on 29 May 2023).
71. ETSI. Experiential Networked Intelligence (ENI). Available online: https://www.etsi.org/committee/1423-eni (accessed on 29 May 2023).
72. Araniti, G.; Iera, A.; Pizzi, S.; Rinaldi, F. Toward 6G Non-Terrestrial Networks. *IEEE Netw.* **2022**, *36*, 113–120. [CrossRef]
73. Zhang, Z.; Xiao, Y.; Ma, Z.; Xiao, M.; Ding, Z.; Lei, X.; Karagiannidis, G.K.; Fan, P. 6G Wireless Networks: Vision, Requirements, Architecture, and Key Technologies. *IEEE Veh. Technol. Mag.* **2019**, *14*, 28–41. [CrossRef]
74. 3GPP. *Solutions for NR to Support Non-Terrestrial Networks (NTN)*; Technical Report TR 38.821; Version 16.2.0; 3rd Generation Partnership Project: Valbonne, France, 2023. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3525 (accessed on 29 May 2023).
75. Shi, J.; Li, Z.; Hu, J.; Tie, Z.; Li, S.; Liang, W.; Ding, Z. OTFS enabled LEO Satellite Communications: A Promising Solution to Severe Doppler Effects. *IEEE Netw.* **2023**, 1–7. [CrossRef]
76. Tomaszewski, L.; Kołakowski, R.; Mesodiakaki, A.; Ntontin, K.; Antonopoulos, A.; Pappas, N.; Fiore, M.; Mosahebfard, M.; Watts, S.; Harris, P.; et al. ETHER: Energy- and Cost-Efficient Framework for Seamless Connectivity over the Integrated Terrestrial and Non-terrestrial 6G Networks. In Proceedings of the Artificial Intelligence Applications and Innovations, AIAI 2023 IFIP WG 12.5 International Workshops, León, Spain, 14–17 June 2023; Maglogiannis, I., Iliadis, L., Papaleonidas, A., Chochliouros, I., Eds.; Springer Nature Switzerland: Cham, Switzerland, 2023; Volume 677, pp. 32–44. [CrossRef]
77. Corici, M.I.; Franke, N.; Heyn, T.; Kontes, G.; Leyh, M.; Magedanz, T.; Maaß, U.; Mikulla, M.; Niemann, B.; Peter, M.; et al. On the Road to 6G: Drivers, Challenges and Enabling Technologies. White Paper Version 1.0, Fraunhofer FOKUS: Berlin, Germany, 2021. Available online: https://cdn0.scrvt.com/fokus/137064883186fe80/9a009606e5a4/6g-sentinel-white-paper.pdf (accessed on 29 May 2023).
78. HEXA-X Project. Gaps, Features and Enablers for B5G/6G Service Management and Orchestration. Project Deliverable D6.1, HEXA-X: A Flagship for B5G/6G Vision and Intelligent Fabric of Technology Enablers Connecting Human, Physical, and Digital Worlds, 2021. Available online: https://hexa-x.eu/wp-content/uploads/2021/06/Hexa-X_D6.1.pdf (accessed on 29 May 2023).
79. Strinati, E.C.; Barbarossa, S.; Gonzalez-Jimenez, J.L.; Kténas, D.; Cassiau, N.; Maret, L.; Dehos, C. 6G: The Next Frontier: From Holographic Messaging to Artificial Intelligence Using Subterahertz and Visible Light Communication. *IEEE Veh. Technol. Mag.* **2019**, *14*, 42–50. [CrossRef]
80. Chen, Y.; Liu, W.; Niu, Z.; Feng, Z.; Hu, Q.; Jiang, T. Pervasive intelligent endogenous 6G wireless systems: Prospects, theories and key technologies. *Digit. Commun. Netw.* **2020**, *6*, 312–320. [CrossRef]
81. *Technical Standard TS 22.186*; Service Requirements for Enhanced V2X Scenarios. Version 17.0.0; 3rd Generation Partnership Project: Valbonne, France, 2022. Available online: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3180 (accessed on 29 May 2023).

82. Tomaszewski, L.; Kołakowski, R.; Zagórda, M. Application of Mobile Networks (5G and Beyond) in Precision Agriculture. In Proceedings of the Artificial Intelligence Applications and Innovations, AIAI 2022 IFIP WG 12.5 International Workshops, Hersonissos, Greece, 17–20 June 2022; Maglogiannis, I., Iliadis, L., Macintyre, J., Cortez, P., Eds.; Springer International Publishing: Cham, Switzerland 2022; Volume 652, pp. 71–86. [CrossRef]
83. European 5G Scoreboard. Available online: https://5gobservatory.eu/observatory-overview/eu-scoreboard/ (accessed on 29 May 2023).

On 6G-enabled SDN-based mobile network User Plane with DRL-based Traffic Engineering*

Robert Kołakowski[1,2] Lechosław Tomaszewski[1] and Sławomir Kukliński[2]

[1] Orange Polska S.A., al. Jerozolimskie 160, 02-326 Warsaw, Poland
[2] Warsaw University of Technology, ul. Nowowiejska 15/19, 00-665 Warsaw, Poland

Abstract. The emerging 6G use cases will pose new challenges for the mobile network User Plane (UP), requiring its rapid evolution in terms of flexibility and intelligent optimisation. To achieve the foremost, the exploitation of the Software-Defined Networking (SDN) concept is commonly considered due to the logically centralised network control and native support for Traffic Engineering (TE). A promising solution to embed intelligence in the network is using Deep Reinforcement Learning (DRL) methods, which are capable of flexible optimisation of complex environments without prior modelling. While there exist several state of the art concepts combining the above technologies pair-wise, there is no approach that integrates them into a unified 6G-ready solution. This paper presents the novel 3GPP-compliant SDN-based UP architecture enhanced by DRL-based TE to facilitate emerging 6[th] Generation (6G) use cases. The approach leverages hierarchical architectures to improve the scalability of operations, support decentralised 6G network deployments and enable DRL usage in carrier-grade mobile networks.

Keywords: beyond 5G · 6G · User Plane · SDN · DRL · AI · UPF · traffic engineering · QoS · user-centricity · TN · NTN

1 Introduction

Software-Defined Network (SDN) is commonly considered an important component for modern telecommunication networks due to forwarding flexibility, fine-grained network traffic control, and facilitation towards application-driven Traffic Engineering (TE) that altogether constitute the overall service agility. Despite the technology potential and increased interest in the SDN market (the growth estimated at 47.3% Compound Annual Growth Rate (CAGR) [1] in years 2020-2031), the network-wide carrier-grade exploitation of SDN in the mobile User Plane (UP) is still at the infancy stage, mainly due to scalability issues.

Simultaneously, the 6G research accelerates. While the standardisation is at a very early stage, with only initial IMT-2030 document releases [2] and the first 3[rd] Generation Partnership Project (3GPP) study forecasted for 2025 [3], the 6G

* ETHER project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101096526. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them.

use cases such as holographic telepresence, Extended Reality (XR), robot/cobot network fabrics, or Non-Terrestrial Network (NTN)-driven applications [4], and 6G features have started to emerge. These will pose new UP requirements in terms of traffic control flexibility and granularity, user/application mobility and resilience. Moreover, intelligent, scalable and automated operation and optimisation are needed to address the rising networks' complexity and contribute to 6G targets. Given these circumstances, a promising solution to implement intelligence in UP is Deep Reinforcement Learning (DRL), which involves intelligent decision-making agents learning via interactions with the environment. Despite successful DRL applications to continuous time-variant networking problems, their usage in carrier-grade networks poses several challenges, such as in-training violations, convergence, coordination of agents, etc.

The goal of the paper is to explore the perspectives and challenges regarding the exploitation of DRL-based TE in 6G SDN-based UP (SDN-UP). First, we present the recent State of the Art (SotA) regarding the 3GPP standardisation (cf. Section 2), Beyond 5G (B5G) SDN-UP and DRL-TE (cf. Section 3) and key challenges regarding DRL exploitation in SDN-UP (cf. Section 4). Based on the analysis, we propose a novel 3GPP-compliant SDN-UP architecture integrated with DRL-based TE, called SUP-DTE (Section 5). Finally, we outline the key benefits of the approach, remaining open issues and solutions (cf. Section 6).

2 User Plane in 5$^{\text{th}}$ Generation (5G) System

The 5G System (5GS) architectural design is based on the fundamental principle of Control and User Plane Separation (CUPS)[5]. In the 5GS Core Network (CN), the user data flow and processing support is nested within the User Plane Function (UPF), which is terminated at Radio Access Network (RAN) node at one side and at the Internet Protocol (IP) data network at the other. Unlike in previous generations UP, UPF can include flexibly designed service-specific user traffic processing chain, e.g., packets inspection, classification, routing, marking, network address translation, etc., formerly supported within the "vestibule" between the anchor point of the user data tunnel and the IP network.

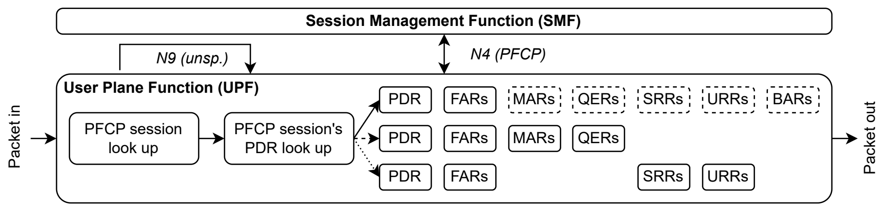


Fig. 1. 3GPP 5G UPF packet forwarding model [6]

UPF embeds the control mechanisms to be exposed to the 5G Control Plane (CP) via the N4 interface [5]. The UPF direct counter-partner in CP is Session Management Function (SMF) (except UPF monitoring events information exposure to other Network Functions (NFs)) acting as a proxy for interactions with CP NFs, e.g., in the procedures of user data session establishment, termination, and modification (also for User Equipment (UE) handovers handling), Quality of Service (QoS) management, etc. For the latter, 5GS supports multiple parameters, which include flow type: non-, ordinary, or delay-critical Guaranteed Bit Rate (GBR), flow priority level, packet delay budget, packet error rate, averaging window, maximum data burst volume. The 5G QoS characteristics based on the above parameters are indicated through 5G QoS Identifier (5QI) classes. Other features are also supported, i.a., allocation and retention priority, reflective QoS, maximum bit rates (per-flow, aggregate) and packet loss rates.

The N4 interface implements the Packet Forwarding Control Protocol (PFCP) used by SMF to control the packet processing within UPF for each individual Protocol Data Unit (PDU) session via PFCP session. For each incoming packet, a look-up of a relevant PFCP session and then of a proper Packet Detection Rule (PDR) among ones provisioned for that session is performed (cf. Fig. 1). When a matching PDR is found, the instruction set is applied to a packet, which may include: Forwarding Action Rule (FAR), Buffering Action Rule (BAR), QoS Enforcement Rule (QER), Usage Reporting Rule (URR), or Multi-Access Rule (MAR). Additionally, a Session Reporting Rule (SRR) can request UPF to detect and report PFCP session events unassociated either with specific PDRs, or traffic usage measurements, e.g., per QoS flow and per UE QoS measurements.

The 5G UP design incorporates some of the SDN concepts (CUPS, packet processing approach, or programmatic control), but it lacks full alignment with SDN and most popular protocol [7], i.e., OpenFlow (OF) [8]. While OF can be used to implement some of the UPF functionalities, including PDR (packet matching to flow entries in flow tables), FAR (actions associated with flow entries), URR and SRR (by per-flow statistics provisioning and configuration of OF switch) there exist deficiencies regarding other rules. Primarily, the OF lacks QER support as it only allows mapping flow to the in-switch queues using identifiers – the queue configuration (priority, scheduling algorithm, maximum rate, etc.) cannot be set by OF. Also, packet delay measurement is not supported, further complicating 5QI enforcement. Finally, as OF lacks buffering control capabilities, implementation of BAR requires external support.

Nonetheless, the SDN can be advantageous for 6G UP. The primary benefits include the logically centralised and programmable control (enabling flexible path establishment and manipulation), the ability to create multiple paths (e.g., for seamless connectivity, increased resilience, etc.), mobility of two path ends (support for UE/application mobility), fast path operations (support for high mobility scenarios, e.g., NTN-based), per-flow granularity (e.g., per-UE), and ability to leverage external information on the network (predicted topology changes, network failures, etc.). Also, the SDN programmability allows for intelligent network optimisation via TE mechanisms, which enable load balancing, minimisation of congestion, E2E delay, packet loss and energy consumption, QoS/Quality of Experience (QoE) maximisation, or optimising resource utilisation. Centralisation, however, reduces scalability and raises the issues of a single point of failure, SDN Controller (SDNC) placement, and SDN nodes' synchronisation. However, the non-unified approach to traffic management and optimisation increases complexity, lowers inter-operability in integrated environments (e.g., using external solutions to connect data centres) and limits possibilities to implement End-to-End (E2E) QoS-aware TE on a per user basis. The E2E TE will require counterparts in network segments belonging to Mobile Network Operator (MNO) and external providers. The latter can implement TE mechanisms conflicting with UP QoS requirements. Such strong coupling can hamper the E2E TE in multi-domain and multi-stakeholder 6G UP.

3 Related work

While for today, the 6G UP definition is vague, the envisioned 6G features can provide insights into potential directions of UP evolution. The International Telecommunication Union – Radiocommunication Sector (ITU-R) released initial materials on the trends for IMT-2030 system [2,9], drafting six usage scenarios, namely, Immersive Communication, Artificial Intelligence (AI) and Communication, Hyper Reliable Low-Latency Communication (HRLLC), Ubiquitous Connectivity, Massive Communication, Integrated Sensing and Communication, falling under the overarching aspects of sustainability, connecting the unconnected, ubiquitous intelligence, security, and network resilience. The IMT-2030 technical performance requirements are expected to appear by Q1 2026 [10]. Nonetheless, academia proposes many UP concepts targeting the above use case categories and core 6G values. The 6G-enabled UP that introduces functional UPF split into CN, RAN, and common sub-functions has been proposed [11]. The E2E UP is formed by chaining elements composed of sub-functions controlled by SMF or Centralised Unit (CU). An "organic network" concept has been presented [12] where CP procedures are grouped into functionalities exposed to the UE and executed as processes. The approach shrinks CP interactions allowing UPF implementation as a traffic control application atop forwarding fabric.

In the context of 6G UP, the incorporation of SDN with a different scope of responsibilities and positioning (SDNC atop or below NFs) is commonly considered. In [13], an SDNC acts as the centralised coordinator of all UPFs (OF-based switches) including Next Generation NodeBs (gNBs), which are connected with OF switches to enable E2E flow management. The Open vSwitch (OVS)-based UPF facilitating network slicing has been proposed [14]. UP slices are created using VLAN tunnels (instead of GPRS Tunneling Protocol (GTP)/IP) leading to lower and more stable latency. The Aether platform introduced Programming Protocol-independent Packet Processors (P4)-UPF [15], in which the SDNC operates below the UPF abstraction and executes commands of the upper-level applications. 3GPP compliance is ensured by PFCP agent acting as the CP termination and UP4 App translating the CP requests into SDN operations. The Intelligent UP (IUP), composed of switches with compute and data processing capabilities enabling In-Network Computing (INC), has been presented [16]. The concept adopts the OF-compliant approach by proposing Communication and Compute Flows to handle network and QoS management on a flow basis.

The autonomous UP optimisation requires intelligent TE. While classical TE in centralised SDN is well addressed [17], the AI-driven TE, especially DRL-based, is still investigated. The DRL-based routing for centralised SDN [18], which considers path-state metrics to adapt to variable traffic conditions, was introduced. The SDN-DRL self-management architecture has been presented [19], which optimises network services to meet QoS demands. Deep Reinforcement Learning Agents (DRLAs) are deployed across network sites, providing routing policies per site for each traffic class using monitoring data. A multi-agent DRL TE framework for a hybrid SDN has been proposed, in which DRLAs generate the routing policy for the SDN switches while legacy routers implement proprietary TE algorithms [20]. The hierarchical multi-controller SDN framework enhanced with TE, called Helix, has been presented [21]. The solution outperforms baseline TE solutions while improving system scalability.

While the above solutions involve the pivotal 6G UP technologies, there exists no approach that combines UP, SDN, and DRL-based TE into a unified E2E framework, while considering the critical issues, such as scalability and flexibility of operations, adaptability or coordination support.

4 DRL-based TE in SDN-UP

DRL is a branch of Machine Learning (ML) combining Reinforcement Learning (RL) and deep learning to provide decision-making for complex environments. A typical DRL setup (cf. Fig. 2) comprises the Deep Neural Network (DNN)-based Agent – the approximator for policy, value or both policy and value functions; and the Environment – the external system the Agent interacts with. The Agent's goal is to learn the optimal policy $\pi : S \times A \to [0, 1]$, maximising the total reward over time. The DRLA learns through feedback by: i) obtaining the information on the current environment state s_t and reward r_t obtained for the last transition; ii) selecting the action a_t based on s_t using π; iii) executing a_t leading environment to transition to s_{t+1} (with reward r_{t+1}). The history of transitions, actions and rewards is used to iteratively improve DRLA policy π.

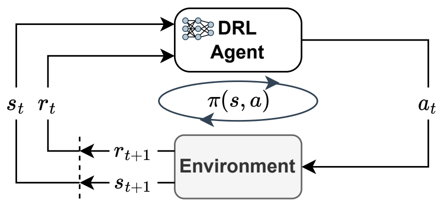


Fig. 2. Typical DRLA setup

While DRLAs are successfully used in multiple fields requiring negotiation or dynamic decision-making (e.g., robotics, healthcare), DRL suffers from several issues, complicating its massive implementation in mobile networks. The key challenges concerning DRL-based TE in SDN-UP are described below.

Sample efficiency: As the DRLA knowledge is built upon experiences, improving sample efficiency to reduce training time is vital [22]. In conventional OF-based SDN setups, the environment state information is obtained via SDN CP channel [8], lowering SDNC's flow handling capacity and limiting the maximum Data Plane (DP) sampling frequency. While particular solutions to improve data efficiency have been proposed, e.g., by increasing DRL model's comprehension of inter-dependencies among states [23], or long-horizon task splitting into shorter ones during training [22], their impact in the SDN context is not assessed, yet.

Reward design: Full flexibility of reward function definition, while highly beneficial in multi-criteria optimisation, requires prior assessment of potential undesired outcomes. In complex constrained environments, such as SDN-UP, the DRLA optimisation of network aspects can lead to the degradation of another Key Performance Indicator (KPI) (e.g., load distribution vs resource utilisation) and lead to potential issues such as decreased fairness across users or prioritisation of certain traffic types. Moreover, the DRL reward is intractable to change – any modification requires the deployment of a DRLA trained using the new reward definition. The promising solutions in the SDN-UP context are meta-learning approaches [24], which can be effectively used to coordinate the training across a set of DRL agents targeting different objectives [25].

Transferability: The DRLA policy (DNN model) is dependent fully on the DRLA setup. As the output policy is optimised in the context of specific Markov Decision Process (MDP) settings, the reusability of the DNN model to solve other MDPs is largely limited. Transferability is essential for SDN-UP to facilitate the creation of DRLAs for specific network segments (e.g., transport) or service types and support coordinated training (e.g., federated learning). A promising actor-critic approach has been proposed [26], in which the network graph embedding is used by a single-input DNN to evaluate routing decisions. A good generalisation for many network topologies allows transferring the critic model.

Generalisation: DRL generalises poorly on samples outside the training distribution, especially for continuous states and actions. Hence, in sensitive environments such as SDN-UP, it is vital to establish a relevant environment model and comprehensive training procedure to mitigate performance degradation during operation. It has been proved that generalisation can be largely improved by using representation learning for states [27] or both actions and states [28].

Robustness: Achieving DRL robustness to faults during long-term optimisation of real-life systems is problematic as the non-stationary environment can destabilise DRLA policy. A recently emerging field of Continual RL studies the aspects of joint agent and environment evolution to mitigate devastating phenomena such as, e.g., catastrophic forgetting [29]. The field is still forming, however, and lacks mature solutions that could be used to improve the DRLAs' robustness to failures, especially in SDN-UP environments.

Observability: MDP requires full observability of the environment states, which is often not feasible in real-life applications. In the SDN-UP, obtaining the actual environment state is impacted by latencies caused, i.a., by the spatial distribution of devices or SDNC placement. A potential solution to this issue involves Partially Observable Markov Decision Process (POMDP) in which the DRLA acts upon a belief state (a probability distribution over possible environment states). The POMDP problems, however, are largely intractable to be solved optimally [30] and suffer from poor convergence.

Multi-stakeholder environment: The 6G network complexity will require vast numbers of intelligent agents to optimise different network aspects. To maintain the stable operation of the network, advanced DRLAs coordination will be vital. Implementation of such mechanisms in a multi-stakeholder environment, e.g., cooperative learning support [31], while considering Service Level Agreement (SLA) constraints, remains a challenge.

While specific solutions have already been proposed to the above-stated challenges, there exists no unified approach supporting their exploitation in SDN-UP. The architecture proposed in Section 5 aims to bridge this gap, as well as address the key SDN-UP and TE issues.

5 Integrated SDN-UP-DRL-TE framework

Hereby, we propose a 6G-enabled SDN-UP with DRL-TE (SUP-DTE) framework, supporting the envisioned multi-domain, multi-stakeholder and 3D character of the 6G network. The architecture design follows the principles proposed by the European Union (EU) ETHER project vision [32]. The cognitive capabilities in SUP-DTE are provided by the DRL-based TE performing network optimisation from both user and network operator perspectives. SUP-DTE comprises five hierarchical frameworks responsible for: i) SDN-based network orchestration; ii) service orchestration; iii) data-path setup and control – 6G User Plane Function (6G-UPF); iv) DRL-TE; v) TE data and DRL-TE models exchange between the network operator and external stakeholders – AI Layer (cf. Fig. 3). Each system operates at both domain and E2E levels.

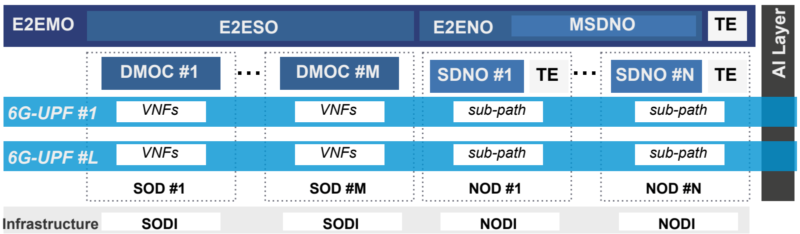


Fig. 3. The high-level view of SUP-DTE components

We consider two types of domains: Service Orchestration Domains (SODs), devoted to the management and orchestration of services, i.e., Virtual Network Functions (VNFs); and Network Orchestration Domains (NODs) devoted to the establishment of connectivity between VNFs (both inter- and intra-SOD). Both NODs and SODs encompass different technological or administrative domains (transport, edge, RAN, etc.) in Terrestrial, Aerial or Satellite network segments and handle the underlying NOD Infrastructure (NODI) and SOD Infrastructure (SODI) resources. The E2E Management and Orchestration (E2EMO) in the system is handled by E2E Service Orchestrator (E2ESO) and E2E Network Orchestrator (E2ENO) that contains Master SDNO (MSDNO) – the supervisor of a federation of SDN Orchestrators (SDNOs). Both E2ESO and E2ENO delegate the UP orchestration processes to domain-level orchestrators Domain Management and Orchestration Components (DMOCs) (e.g., ETSI MANO [33]) and SDNOs (cf. Fig. 5).

In SUP-DTE, the UP is handled by a 6G-UPF, which exploits the MSDNO/SDNO interfaces to establish, maintain and reconfigure network paths (and their parts in NODs, further called *sub-paths*) according to the session-level configuration provided by the network CP (i.e., SMF). Both 6G-UPFs and SDN orchestration components are integrated with the DRL-based TE to allow concurrent and coordinated network and path optimisation. To make UP *user-centric*, 6G-UPF instances are deployed for individual PDU sessions. 6G-UPF implements a hierarchical structure (cf. Fig. 4) in which, master-UPF (mUPF) controls a set of sub-UPFs (sUPFs) to enforce path configuration on both E2E and domain levels. To ensure compliance with the 3GPP QoS model, mUPF contains the *PFCP agent*, which terminates the N4 interface and translates PFCP messages and session configuration, i.e., Flow Processing Rules (XXRs), to atomic UP operations. The CP commands are delivered by sub-UPF Manager (sUPF-M) to respective sUPF managers, i.e., sub-UPF Agent (sUPF-A), and MSDNO for execution. The foremost occurs for intra-domain UE mobility. On CP request, sUPF-M sends path-related commands to the sUPF-A, which executes them via SDNO interfaces. In the case of inter-domain mobility, sUPF-M triggers the E2ENO processes to both reconfigure the E2E path as well as orchestrate sUPFs to handle newly established sub-paths. sUPF-M is also responsible for the aggregation of sub-paths KPIs provided by sUPF-A to obtain E2E data and its verification against PFCP-driven session configuration (sUPF-M will trigger relevant reconfiguration actions in case of violations).

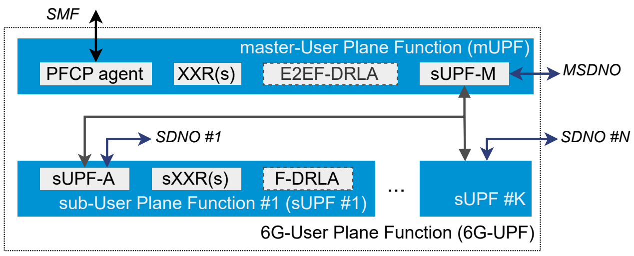


Fig. 4. 6G-UPF internals and interconnection with its sub-components, i.e., sUPFs

The E2E path establishment process is as follows. When UE initiates PDU establishment procedure, the E2ESO orchestrates mUPF in one of the managed SODs. The sUPF-M requests from the MSDNO to establish the abstracted E2E path composed of NODs and SODs Border Nodess (BNs) – i.e., gateways to NODs/SOD or nodes connected to the data source/sink – with session-specific requirements (QoS class, additional UP functions such as in-datapath analytics, firewalls, etc.). MSDNO splits the E2E path into sub-paths (based on the NOD membership) and delegates their orchestration to relevant SDNOs (including the information on the UP VNFs that have to be included in the sub-path). The SDNOs interact with SDNCs (SDN, or Software-Defined Wide Area Network (SD-WAN) in WAN Infrastructure Manager (WIM) case) or relevant brokers if the operator does not own the underlying infrastructure. When the path is established, E2ESO orchestrates sUPF for each sub-path (via DMOCs), with sub-paths metadata, privileges to connect to SDNO and NOD TE framework as well as the data needed to register each sUPF in mUPF.

The configuration for mUPF/sUPF is stored within the XXRs/sub-XXRs (sXXRs) – corresponding to 3GPP rules, e.g., URR, QER. sXXRs is a set of individual rules to be enforced by sUPFs, which contribute to overall XXR (e.g., GBR, delay). To surpass the SDN limitations described in Section 2, we envision the enhanced SDNC, which, in addition to OF, implements parallel interfaces enabling dynamic queue and buffering management (e.g., in the case of OVS, connection to switch OVS Database (OVSDB) server). Moreover, sUPF-M and sUPF-As can interact with E2ENO and SDNOs to collect network-level metrics and suggest optimisation actions to improve QoS, based on the in-built optimisation engines E2E Flow-DRLA (E2EF-DRLA)/Flow-DRLA (F-DRLA). To this end, the 6G-UPF can be seen as the distributed agent (with sub-path granularity) responsible for user data path monitoring and management in case of QoS violations or mobility events. To decrease resource footprint, the 6G-UPF entities can be instantiated as lightweight software components (e.g., containers) or processes within MSDNO/SDNO frameworks. The 6G-UPF deployment within SUP-DTE architecture is depicted in Fig. 5.

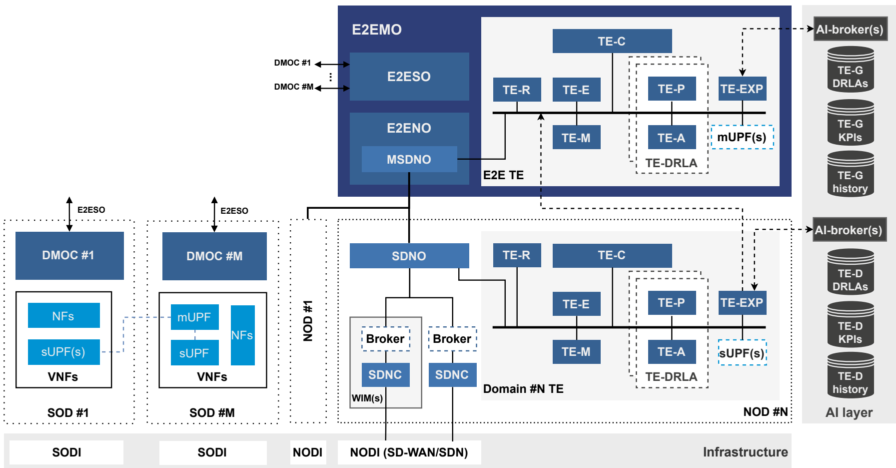


Fig. 5. The SUP-DTE architecture with 6G-UPF instances deployed in SODs and connected to TE frameworks message buses (marked with dashed line)

In SUP-DTE, the network-level UP optimisation is performed by the E2E and domain TE frameworks – meshes of services devoted to network monitoring, analytics, decision-making and enforcement, data exposure or supplementary functions. The TE services are connected to the message bus and communicate using the publish/subscribe paradigm to synchronise actions across services (e.g., maintain common information state and network KPIs). Also, the TE services compose multiple parallel Monitor-Analyse-Plan-Execute based on Knowledge

(MAPE-K) loops [34] concurrently optimising the network. The TE service types are described below.

TE Monitoring Service (TE-M) – responsible for network resources monitoring (physical or virtual links) managed by SDNO/MSDNO, assembling common environment state information (e.g., link utilisation, network topology, etc.) and KPIs calculation for both infrastructure and flows.

TE Analytics Service (TE-A) – providing analytics using the TE-M data, e.g., state representation recalculation, calculation of rewards for the DRL agents, flow- and network-related predictions, etc.

TE Planning Service (TE-P) – responsible for deriving optimisation decisions based on the analytics data regarding flows and network reconfiguration. The DRLAs comprise both TE-A and TE-P functionality.

TE Coordination Service (TE-C) – responsible for the coordination of actions proposed by DRLAs, TE-Ps and sUPFs/UPFs connected to the Domain/E2E TE frameworks.

TE Execution Service (TE-E) – translating commands/policies selected by TE-C into SDNO/MSDNO requests and their execution.

TE Reconfiguration Service (TE-R) – component exposing the TE framework to enable reconfiguration of TE components or network – i.e., Human In The Loop (HITL) – to mitigate instabilities or improve the DRLAs performance.

TE Exposure Service (TE-EXP) – a gateway to the TE framework enabling: i) connection with AI-brokers for DRLA models selection to use within TE framework and network data/KPIs/reconfiguration history monetisation/purchase; ii) network data and KPIs exposure to the E2E TE framework. Due to different dynamics and scope, we assume dedicated data and model databases for Domain-level TE (TE-D) and Global-level TE (TE-G). The aspects of charging and secure data/model exposure are out of TE scope.

It must be noted that NOD and E2E TE frameworks operate with different time scales and can target conflicting optimisation objectives. Whereas the domain TE uses the link-level metrics and performs optimisation in a fast manner, it is assumed that the E2E TE focuses on NOD and metrics abstractions and slow-scale optimisation.

6 Benefits and open issues

The core SUP-DTE benefits in the context of 6G UP include:

- **User-centricity** – 6G-UPFs is focused on individual session, which allows for *user-centric* QoS management and rapid path reconfiguration in case of QoS degradation. Also, the fine-grained QoS monitoring (sub-path and flow granularity) is provided, which enables E2E path fine-tuning.
- **Service chaining** – SDN enables flexible path split to include in-datapath VNFs to form service chains or perform INC. Moreover, the 6G-UPF abstraction eliminates the need for GTP tunnelling to reduce related DP overhead.
- **Autonomic optimisation** – using DRLAs both at the network and user-session level facilitates autonomous QoS-aware network-wide optimisation.

The TE framework can include information on the flow QoS requirements and constraints into optimisation processes.
- **Lightweightness** – the 6G-UPF lifespan equals the session duration limiting the static resource consumption during UE inactivity periods. The dynamic orchestration of 6G-UPF instances, however, will lead to increased resource consumption by Management and Orchestration (MANO). Hence, it would be necessary to empirically derive optimal 6G-UPF lifespan to mitigate resource overspending on its life cycle management operations.
- **Scalability** – hierarchical approach improves the scalability of SDN (due to SDN CP distribution) and DRL-TE (due to reduced network state sizes).
- **DRL-TE support** – the hierarchical distributed architecture contributes to: i) faster DRLAs convergence due to better observability of network (more accurate and up-to-date monitoring data due to the collocation of SDNC and DP devices) and reduced state sizes; ii) contribution to data efficiency problem due to more frequent DP sampling (i.e., smaller impact on the SDN CP) and sampling adaptation to the NOD specifics; iii) improved transferability of the DRL models across the same domain types.
- **Cooperative learning and coordination** – the SDNO acts as the umbrella abstracting the resources of SDN/SD-WAN providers. Such an approach enables the implementation of network-wide cooperative learning in TE, e.g., multi-agent DRL, without considering the provider-level optimisation mechanisms. Moreover, the inclusion of mUPF/sUPF into the reconfiguration process allows for improved optimisation decisions while conforming to session-level restrictions.

While SUP-DTE provides multiple benefits regarding the 6G UP implementation, several open issues remain to be addressed. The critical issues, together with potential solutions, are listed below.

Country-wide deployments – while the proposed hybrid approach significantly improves the scalability of SDN and TE, centralised E2ENO might not be scalable enough to accommodate millions of active UEs in country-wide deployments. The potential solution would be to create a mesh of interconnected SUP-DTE responsible for handling the traffic in certain parts of the country and higher-level supervising components to implement the E2E behaviour and maintain E2E QoSs.

Network abstraction – hierarchical approach imposes the creation of the network and KPIs abstractions that provide an insightful view of the actual network state. While network graph abstraction approaches are well-studied [35], their application and efficiency in SDN context have not been well explored yet and remain a serious concern.

UE mobility handling – inter-domain UE mobility might require the reconfiguration of the 6G-UPF, migration of mUPF/sUPF across SODs or deployment of additional sUPF. Frequent handovers can result in large resource overhead regarding E2EMO operations.

DRLAs coordination – concurrent deployment of several DRLAs optimising various conflicting objectives can potentially result in poor algorithms' conver-

gence and performance. To this end, effective coordination mechanisms have to be developed to verify the actions proposed by DRLAs and introduce the reconfiguration prioritisation to allow reaching key optimisation objectives.
SLA maintenance – the DRLA training can lead to severe SLA degradation. Therefore, it is needed to provide a relevant representation of the UP environment for the DRLAs pre-training and evaluation, i.e., UP Digital Twins (DT), before deployment in the production network. Moreover, the supporting solutions that control the agents' behaviour and implement corrective actions in case of instabilities have to be instantiated.
Environment state representation – the integration with NTNs requires the creation of network state representations that encapsulate the aspects of time-variant network topology, including link/node activity and dynamic topology changes. In distributed networks, this issue is aggravated by the cross-domain mobility of nodes and the resulting variable size of the network states.

Overall, there is a lack of comprehensive approach to DRL-TE in SDN-UP. While there exist solutions that address specific networking problems, there are no "Commercial Off-The-Shelf (COTS)" TE solutions that could be effectively applied to carrier-grade deployments. The SUP-DTE approach aims to facilitate the development and integration of TE solutions by introducing a hierarchical distributed architecture, service-based TE framework, and integration with external AI data and model providers.

7 Summary and conclusions

In this paper, we analysed the current SotA regarding key 6G UP components, namely, SDN, 3GPP NFs and DRL-TE, with respect to architectural features, challenges, open issues and their integrations. On that basis, we propose the novel 6G-enabled UP architecture called SUP-DTE, which integrates SDN, 6G-UPF, DRL-based TE and MANO into a hierarchical 6G-ready UP architecture. The key benefits of the SUP-DTE include *user-centric* data-path provisioning, improved operations scalability, embedded intelligent optimisation and support for DRL-based TE, and reduced UPF resource footprint. Finally, we present the remaining open issues to be addressed to enable the implementation of fully autonomous, flexible and intelligent UP network-wide.

References

1. Business Research Insights: SDN orchestration market size, share, growth, and industry analysis, by type (solutions, & services), by application (cloud service providers, telecom service providers, & others) and regional forecast to 2031. [Website] (Jan 2024), `https://www.businessresearchinsights.com/market-reports/sdn-orchestration-market-109427`, Last accessed 09 Feb 2024
2. ITU-R: Future technology trends of terrestrial International Mobile Telecommunications systems towards 2030 and beyond. Report M.2516-0 (11/2022), International Telecommunication Union – Radiocommunication Sector (Nov 2022)
3. Huawei: 6G: The next horizon from connected people and things to connected intelligence. White paper, Huawei: Shenzen, China (2021), `https://www-file.huawei.com/-/media/corp2020/pdf/tech-insights/1/6g-white-paper-en.pdf?la=en`

4. NGMN: 6G use cases and analysis. White Paper ver. 1.0, Next Generation Mobile Networks (NGMN) Alliance (Feb 2022), https://www.ngmn.org/wp-content/uploads/NGMN-6G-Use-Cases-and-Analysis.pdf
5. 3GPP: System architecture for the 5G System (5GS). Technical Standard TS 23.501, ver. 18.5.0, 3rd Generation Partnership Project (Mar 2024)
6. 3GPP: Interface between the Control Plane and the User Plane nodes. Technical Standard TS 29.244, ver. 18.5.0, 3rd Generation Partnership Project (Mar 2024)
7. Yadav, R., Kamran, R., Jha, P., Karandikar, A.: Applying SDN to mobile networks: A new perspective for 6G architecture (2023), doi: 10.48550/arXiv.2307.05924
8. ONF: OpenFlow switch specification; version 1.5.1 (protocol version 0x06). Specification ONF TS-025, Open Networking Foundation (Mar 2015), https://opennetworking.org/wp-content/uploads/2014/10/openflow-switch-v1.5.1.pdf
9. ITU-R: Framework and overall objectives of the future development of IMT for 2030 and beyond. Recommendation M.2160-0 (11/2023), International Telecommunication Union – Radiocommunication Sector (Nov 2023)
10. ITU-R: IMT towards 2030 and beyond. [Website] (2024), https://www.itu.int/en/ITU-R/study-groups/rsg5/rwp5d/imt-2030/Pages/default.aspx, Last accessed 20 Feb 2024
11. Cha, J., Moon, Y., Cho, S., Kim, D., Choi, J., Jung, J., Lee, J., Choi, S.: RAN-CN converged user-plane for 6G cellular networks (2022), doi: 10.1109/GLOBECOM48099.2022.10001487
12. Corici, M., Troudt, E., Magedanz, T., Schotten, H.: Organic 6G networks : Decomplexification of software-based core networks. In: 2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit). pp. 541–546 (2022), doi: 10.1109/EuCNC/6GSummit54941.2022.9815730
13. Abdulghaffar, A., Mahmoud, A., Abu-Amara, M., Sheltami, T.: Modeling and evaluation of software defined networking based 5G core network architecture. IEEE Access 9, 10179–10198 (2021), doi: 10.1109/ACCESS.2021.3049945
14. Costa-Requena, J., Poutanen, A., Vural, S., Kamel, G., Clark, C., Roy, S.K.: SDN-based UPF for mobile backhaul network slicing. In: 2018 European Conference on Networks and Communications (EuCNC). pp. 48–53 (2018), doi: 10.1109/EuCNC.2018.8442795
15. Open Networking Foundation: Using P4 and programmable switches to implement a 4G/5G UPF in Aether. [Website] (Sep 2021), https://opennetworking.org/news-and-events/blog/using-p4-and-programmable-switches-to-implement-a-4g-5g-upf-in-aether/, Last accessed 10 Apr 2024
16. Schwarzmann, S., Trivisonno, R., Lange, S., Civelek, T.E., Corujo, D., Guerzoni, R., Zinner, T., Mahmoodi, T.: An intelligent user plane to support in-network computing in 6G networks. In: ICC 2023 - IEEE International Conference on Communications. pp. 1100–1105 (2023), doi: 10.1109/ICC45041.2023.10279652
17. Abbasi, M.R., Guleria, A., Devi, M.S.: Traffic engineering in software defined networks: A survey. Journal of Telecommunications and Information Technology 2016(4), 3—14 (Dec 2016), https://jtit.pl/jtit/article/view/757
18. Kim, G., Kim, Y., Lim, H.: Deep reinforcement learning-based routing on software-defined networks. IEEE Access 10, 18121–18133 (2022), doi: 10.1109/ACCESS.2022.3151081
19. Altamirano, J.C., Slimane, M.A., Hassan, H., Drira, K.: QoS-aware network self-management architecture based on DRL and SDN for remote areas. In:

2022 IEEE 11th IFIP International Conference on Performance Evaluation and Modeling in Wireless and Wired Networks (PEMWN). pp. 1–6 (2022), doi: 10.23919/PEMWN56085.2022.9963841
20. Guo, Y., Tang, Q., Ma, Y., Tian, H., Chen, K.: Distributed traffic engineering in hybrid software defined networks: A multi-agent reinforcement learning framework (2023), doi: 10.48550/arXiv.2307.15922
21. Zaicu, N.F., Luckie, M., Nelson, R., Barcellos, M.: Helix: Traffic engineering for multi-controller SDN. In: Proceedings of the ACM SIGCOMM Symposium on SDN Research (SOSR). pp. 80—87. SOSR '21, Association for Computing Machinery, New York, NY, USA (2021), doi: 10.1145/3482898.3483354
22. Feng, W., Han, C., Lian, F., Liu, X.: A data-efficient training method for deep reinforcement learning. Electronics 11(24) (2022), doi: 10.3390/electronics11244205
23. Wang, Z., Jiang, M.: Enhancing data efficiency in reinforcement learning: a novel imagination mechanism based on mesh information propagation (2023), doi: 10.48550/arXiv.2309.14243
24. Bing, Z., Lerch, D., Huang, K., Knoll, A.: Meta-reinforcement learning in non-stationary and dynamic environments. IEEE Transactions on Pattern Analysis and Machine Intelligence 45(3), 3476–3491 (2023), doi: 10.1109/TPAMI.2022.3185549
25. Zhang, Z., Wu, Z., Zhang, H., Wang, J.: Meta-learning-based deep reinforcement learning for multiobjective optimization problems (2022), doi: 10.48550/arXiv.2105.02741
26. Almasan, P., Suárez-Varela, J., Rusek, K., Barlet-Ros, P., Cabellos-Aparicio, A.: Deep reinforcement learning meets graph neural networks: Exploring a routing optimization use case. Computer Communications 196, 184–194 (2022), doi: 10.1016/j.comcom.2022.09.029
27. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms (2017)
28. Zhao, T., Wang, Y., Sun, W., Chen, Y., Niub, G., Sugiyama, M.: Representation learning for continuous action spaces is beneficial for efficient policy learning (2022), doi: 10.48550/arXiv.2211.13257
29. Abel, D., Barreto, A., Roy, B.V., Precup, D., van Hasselt, H., Singh, S.: A definition of continual reinforcement learning (2023), doi: 10.48550/arXiv.2307.11046
30. Monahan, G.E.: A survey of partially observable Markov decision processes: Theory, models, and algorithms. Management Science 28(1), 1–16 (1982), `http://www.jstor.org/stable/2631070`
31. Ibrahim, A.M., Yau, K.L.A., Chong, Y.W., Wu, C.: Applications of multi-agent deep reinforcement learning: Models and algorithms. Applied Sciences 11(22) (2021), doi: 10.3390/app112210870
32. Tomaszewski, L., Kołakowski, R., Mesodiakaki, A., Ntontin, K., Antonopoulos, A., Pappas, N., Fiore, M., Mosahebfard, M., Watts, S., Harris, P., Lin, C.K., Santiago, A., Lazarakis, F., Chatzinotas, S.: ETHER: Energy-and cost-efficient framework for seamless connectivity over the integrated terrestrial and non-terrestrial 6G networks (Jun 2023), doi: 10.1007/978-3-031-34171-7_2
33. ETSI: Network Functions Virtualisation (NFV); Management and Orchestration. Group Specification, GS NFV-MAN 001 V1.1.1, European Telecommunications Standards Institute (Dec 2014), `https://www.etsi.org/deliver/etsi_gs/nfv-man/001_099/001/01.01.01_60/gs_nfv-man001v010101p.pdf`
34. IBM: An architectural blueprint for autonomic computing. IBM Autonomic Computing White Paper, Fourth Edition (Jun 2006)
35. Zhou, F.: Methods for network abstraction. Ph.D. thesis, University of Helsinki, Finland (2012)

Hybrid SDN-Based Data-Centric Mobile Core Network with Content-Based Routing

Robert Kołakowski*◇, Thierry Lejkin†, Victorien Romain†‡

*Orange Polska, ◇Warsaw University of Technology, †Orange Innovation, ‡Conservatoire National des Arts et Métiers (CNAM)

***Abstract*—The evolution of 6G core network architecture is a critical research area requiring comprehensive investigation and collaborative efforts. While multiple divergent visions exist, there is a consensus that 6G should be built on 5G achievements, such as softwarization, Service-Based Architecture (SBA), or network slicing, to ensure economic sustainability and foster innovation. Yet, most of the 6G architectural visions are revolutionary, raising questions about their feasibility in terms of financial sustainability and practical deployment. This paper introduces a Hybrid SDN-based Data-Centric Core Network (HSDCN) architecture combining multi-controller Software-Defined Networking (SDN), data-centric principles, and content-based routing to enable an optimized core network in cloud-native environments and facilitate the evolution of 5G networks towards 6G. HSDCN is a hybrid mobile network architecture, supporting low-latency Control Plane (CP) operations and facilitating AI-driven network operations while maintaining compatibility with 3GPP standards and avoiding disruptive design shifts.**

***Index Terms*—6G, data-centric, core network, multi-controller SDN, content-based routing, AI**

I. Introduction

As 5th Generation (5G) reaches maturity, research and standardization bodies are exploring 6th Generation (6G) use cases and evolution directions. There is a consensus that 6G networks should ensure environmental and economic sustainability [1], [2]. To meet these goals cost-effectively, 6G is expected to reuse 5G components while adding new features to deliver services and value to customers [1]. Hence, 6G should enable continuous and incremental evolution by design [2]. Yet, many proposals for the 6G core network design are revolutionary, contradicting the above aims.

To leverage the cloud ecosystem, mobile Core Network (CN) architecture requires several improvements, including inter-Network Function (NF) communication efficiency [3], lighter NFs (evolution towards microservice architecture) [4], or overall organization of data flow [5]. A data-centric paradigm can facilitate this evolution by introducing stateless NFs and reducing NF interactions to operations on data, thereby increasing flexibility and simplifying NF implementation and upgrades [5], [6]. Frequent database access and synchronization delays, however, can increase CP latency [7], which can be critical for procedures requiring prompt execution (e.g., satellite-based) and complicates database design.

Software Defined Network (SDN) is considered one of the key enablers to provide programmability in 6G mobile networks. The logical centralization and software-driven control enable optimization and advanced routing, e.g., multi-path or content-based routing [8]. For today, SDN is mainly considered in the User Plane context while its potential for exploitation in CN is often overlooked.

In this paper, we introduce the Hybrid SDN-based Data-Centric Core Network (HSDCN) architecture, which extends 3GPP 5G CN with mechanisms supporting gradual evolution towards 6G. In HSDCN, we introduce SDN-driven content-based and multi-path routing capabilities for CN exchange customization, including complex NF chaining, parallelization, or support for data-centric and AI-native networks. The proposal requires minimal changes to the implementation of 3GPP NFs.

The structure of the paper is as follows: Section II presents related works, Section III describes the HSDCN concept, Section IV presents its simple evaluation, and Section V summarises and concludes the paper.

II. Related Work

The 5G System (5GS) marked a paradigm shift in mobile networks through network softwarization and virtualization [9], non-monolithic architecture with modular NFs [9], or Service-Based Architecture (SBA). Today, 5GS is highly complex featuring over 40 NFs [9] and over 500 procedures [10]. Although 6G standardization is in its initial stages, there is a consensus pointing to reusing 5GS components while pursuing simplification, flexibility, and financial sustainability [2]. So far, the generic 6G requirements have been outlined, focusing on, i.a., improvement of network integration capabilities & flexibility, Artificial Intelligence (AI)-native design, true cloud-nativeness, or automation [2], [11], [12]. New usage scenarios have also been identified [11], such as Immersive/Massive Communication, or Ubiquitous Connectivity.

To address the above, several revolutionary 6G architecture visions have emerged. The key proposed paradigms include the user-centric (per-UE) mobile networks [13], decomposing NFs into μNFs following IT microservice architectures [4], parallelizing procedure execution [3], stateless NF and data-centric [5], [14] network or procedure-oriented network [15]. The latter proposes to implement End-to-End (E2E) procedures with stateless NFs, to be executed by generic workers on requests issued via CP front-ends. Finally, AI architectures integrating AI with 6G, such as DAEMON [16], are proposed.

Migration towards truly cloud-native CN involves decoupling of data and its processing. It can be achieved via data-centric architectures comprising stateless NFs [5], [14]. In these approaches, NFs execute procedures by modifying the User Equipment (UE)/network contextual information in a dedicated network database, reducing complex inter-NF interactions to sequential operations on data. The benefits include, e.g., lowered complexity, enhanced security, or better quality of AI/ML models [14]. The frequent database access, however, can increase CP delay. Another obstacle is the definition of SBA and Service Communication Proxy (SCP) [5] – a centralized CP message broker for NF clusters. SCP aggregates CP messages prior to routing, which constitutes a potential choke point for CP links, also impacting the CP delay (resulting in longer processing at SCP). Also, the 5G SBA remains host-centric (NFs mapped to IP locators), limiting scalability and suitability for distributed NF deployments [5].

A solution to these issues is to deliver CP traffic concurrently to databases and NFs via Service Function Chaining (SFC) [17] for traffic flow processing and policy-based routing. The traditional IETF SFC architecture is protocol-agnostic and assumes unicast interactions between nodes, which poses limitations on chain configuration. State-of-the-art approaches implement SFC using various mechanisms and protocols, such as Segment Routing (SR) [18] . More flexible approaches for chaining that leverage content-based routing have been proposed. Using OpenFlow (OF) and Virtual LAN (VLAN) tagging for content-based routing was introduced in [19]. A network layer content-based publish-subscribe mechanism based on P4 switches is proposed in [8]. In [20], a protocol-oblivious content-based routing architecture using SDN is proposed, showing SDN's suitability for content-centric networks.

Most 6G architectures are disruptive and costly. To ensure sustainable evolution, 6G should build on 5G through iterative upgrades for rapid monetization. We propose a CN architecture combining distributed SDN, data-centricity, and content-based routing to bridge these gaps.

III. Hybrid SDN-based Data-Centric Core Network (HSDCN)

The main goals of HSDCN are to enhance CN flexibility, facilitate iterative CN evolution and low-cost adaptation. To this end, HSDCN considers: i) support for data-centric networking; ii) CP traffic optimisation; iii) using SFCs to implement procedures to optimize CP latency; and iv) integration with AI functions and frameworks. The concept reuses 3rd Generation Partnership Project (3GPP) NFs definition [9] with minimal extensions to facilitate implementation of new services based on already implemented NFs. HSDCN comprises three interworking parts described in the next sections (cf. Fig. 1):

- *Hierarchical Multi-controller SDN* – responsible for management and control of the CP network at the transport level. It comprises components and APIs supporting content-based and multi-path routing for CP NFs;
- *Data-centric CN* – comprising legacy 5G and 6G Intelligent NFs (INFs) – the functions with embedded AI capabilities. The NFs are either stateless or stateful instances and implement traditional REpresentational State Transfer (REST)-based interfaces for communication over SBA. Each NF/INF can implement Content-based Routing Proxy (CBRP), which is an additional interface that provides content-based routing capabilities in CN. The CBRP behaviour is handled by SDN controllers via dedicated Application Programming Interfaces (APIs);
- *Intelligence Plane (IPL) components* – comprising AI-driven functions, both generic and network-oriented ones, and database systems for centralized data storage acting as a single source of truth for network operations.

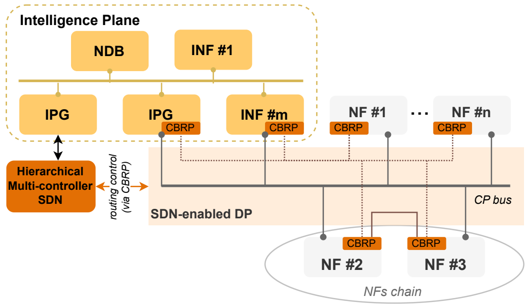


Fig. 1: Overall view of HSDCN concept

A. Hierarchical multi-controller SDN

The proposed hierarchical multi-controller (multi-domain) SDN aims to provide optimized connectivity for the signaling traffic and increase the flexibility of inter-NF exchange at scale via content-based routing and multi-path support (cf. Fig. 2). The framework comprises two levels: i) *global level* – centralized Global SDNC (G-SDNC), which is responsible for inter-domain routing and global network control. G-SDNC enforces inter-domain routing policies by instructing domain-level SDN Controllers (SDNCs) to install forwarding rules in Data Plane (DP) devices. It is assumed that G-SDNC can have a limited network view to hide the network complexity. In such a case, the minimal view comprises the network graph composed of NF deployment sites (e.g., data-centers), domain gateways (nodes that interconnect different domains), and abstracted connections between them (if the nodes are reachable); ii) *domain level* – responsible for the control and management of connected DP devices (e.g., OF switches); Due to different requirements associated with the spatial distribution of equipment, we consider two domain types: Local SDNC (L-SDNC) referring to LAN and MAN SDN networks, and WAN SDNC (W-SDNC) – Software-Defined Wide Area Networks (SD-WANs) interconnecting remote sites.

Each NF can be extended with Content-based Routing Proxy (CBRP), whose role is two-fold. First, it filters the outbound traffic based on the predefined Content Filter (CF) set by SDN components. This behavior can be implemented either as a proxy to the NF REST interface, via scanning of outbound traffic (which is costly in terms of latency), or within the interface implementation (a latency-efficient option). Second, it performs the packet marking by injecting unique tags into the packets that match the defined filter. The tags can be injected directly into the packet headers (e.g., DSCP/TOS fields, IPv6 extension headers) or via encapsulation (VLAN, VXLAN, etc.). The DP devices must be capable of matching

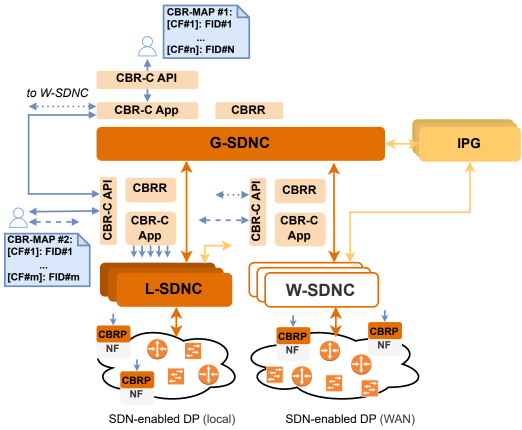


Fig. 2: Content-based routing support in HSDCN

on the fields to enable packet forwarding based on the tag set in packet header. Each CF is associated with a unique Flow Identification Data (FID) that comprises:

- Flow Graph (FG) – a directed graph containing NF as vertices and links symbolizing the connections representing data flow to implement;
- unique tag to be associated with the flow installed in the network devices (e.g., VLAN ID), and;
- discriminatory features (primarily at Layer 7), such as, e.g., NF service types, NF service groups (e.g., authorization, authentication), NF endpoints, network events, or values within messages (configurable by operator).

At once, CBRP can run multiple CFs simultaneously. The combination of FID and CF enables the implementation of a low-overhead publish-subscribe mechanism using simple SDN routing. Moreover, the pure tag-based forwarding is enabled without the need to specify IP addresses for either the packet source or destination. As a result, this approach contributes to the cloud-nativeness of the architecture as the NF interactions are defined by the data flow between labeled NFs rather than IP source-destination pairs (diminishing the burden of IP address pools management). It must be emphasized that setting a tag-based processing does not necessitate resigning from traditional Internet Protocol (IP)-based routing. It is required, however, to reserve tag range for CBRP to avoid potential conflicts with other SDN forwarding methods. The E2E content-based routing is set up by the following components:

- *Content-based Routing Configuration (CBR-C) API (frontend):* the interface for programmatic configuration of content-based routing at global and domain levels; the external parties can request the configuration by providing the map comprising a list of one or more CFs with assigned FIDs. After configuration validation, the map is sent to the CBR-C App for enforcement.
- *Content-based Routing Configuration (CBR-C) App:* SDN application responsible for enforcing the configuration of content-based routing and calculating the actual paths between NFs specified within FG. At the global level, upon configuration request, CBR-C App calculates rough E2E paths for the entire FG. The rough paths comprise compute nodes in which the NFs are deployed and domain gateways. Then, CBR-C App splits the paths based on domain membership of nodes to form FG for each domain, and sends them with the remaining FID configuration (discriminatory data, VLAN ID, etc.) to relevant CBR-C Apps for enforcement (via CBR-C APIs). At the domain level CBR-C App computes the path for the FG and requests SDNC to install entries in the DP switches that contain matching instructions on defined tags and forwarding rules. Finally, CBR-C App updates the CBRP configuration of switches with CF and FID, establishing an E2E path (cf. Fig. 4).
- *Content-based Routing Registry (CBRR):* registry which stores the content-based routing configuration, including the maps of CF and FID installed in CBRP, active rules, security policies, etc.;

Each SDN controller also maintains the connection with the Intelligence Plane (IPL) via Intelligence Plane Gateways (IPGs) (see Section III-B) to store network Key Performance Indicators (KPIs), e.g., links utilization, delay, etc., and consume services provided by AI functions to optimize traffic flows between CN NFs. Finally, while we consider a wide-scale adoption of SDN, exploitation of a hybrid network comprising also non-SDN capable domains can be used as long as preconfigured tag-based forwarding is supported by the non-SDN capable switches (e.g., DSCP).

B. Intelligence Plane & data-centric Core Network (CN)

We extend the 5G CN with components supporting Intelligence Plane (IPL), data-centricity, AI-driven operations and content-based routing (cf. Fig. 3).

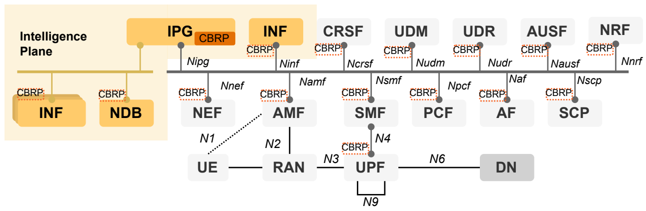


Fig. 3: Data-centric enabled CN and interactions with IPL

The IPL is a vertical network plane that spans all other network planes (e.g., Control Plane, User Plane, Management Plane), aggregates intelligent AI-based functions deployed at any network layer, and provides logically centralized data storage and exposure. It comprises: (cf. Fig. 3):

- *Intelligent NF (INF):* a generic function embedding cognitive capabilities. INF can be either the 3GPP NF with embedded AI model (i.e., native-AI NF) or a generic engine implementing an AI/ML algorithm and exposing AI services to external consumers (e.g., analytics).
- *Intelligence Plane Gateway (IPG):* the entrypoint to the IPL service mesh, providing service discovery capabilities for NF and INF and admission control;
- *Core Network Database (NDB):* the physically distributed and logically centralized database (data lake) storing

network-related data (network/user states, contexts, etc.) and providing system-wide synchronized access to it. By design, Core Network Database (NDB) instances are expected to be collocated, with CN functions for high-availability access. As the storage functionality will span multiple network domains, the NDB should employ data retention, replication, and synchronization mechanisms to maintain a globally consistent network state information;

Each IPL component can communicate either over the IPL message bus (publish/subscribe) or use the content-based routing mechanism. Regarding data-centric core, we consider both stateless and legacy stateful NFs. The foremost utilize NDB services (accessed via IPG) to store, collect, and update state data during procedure execution. For legacy NFs, additional mechanisms for replicating data stored within 5G NFs should be adopted (e.g., UDM, UDR, or NRF content mirroring) to maintain a globally consistent network state information.

Regarding state definition in NDB, the compromise strategy described in [14] should be adopted, which proposes user and NF type granularity – one state for each user dedicated to a specific network function type. The state storage by NFs can be implemented in various ways, i.e., at the message level (state construction at the initial procedure step and consecutive updates during the next steps), transaction level (creation of state after each transaction), and procedure level (construction after all transactions are executed) [14]. A message-level approach minimizes latency impact during operation and facilitates faster failure recovery. Hence, every time the CN procedure is executed, the respective contextual information for the UE/NF is created/updated in NDB.

As reusing the existing 5G NFs is of top importance in terms of economic savings, 3GPP NF REST interfaces can be extended with CBRP (see Section III-A) for configuration of tags associated with specific message contents and/or services of NF or NF group (e.g., security, authentication) to enable fine-grained routing of CP messages. Moreover, we propose an additional NF called Content-based Routing Support Function (CRSF) (a type of Application Function (AF)), which extends the role of SCP in terms of service discovery and additionally acts as the interconnect point between the NFs and content-based SDN applications residing within the SDN application plane. It enables NFs to request service subscriptions via content-based delivery without direct exchange with the NF.

Content-based and multi-path routing can be used to implement SFCs, as shown in Fig. 4. NFs can be chained to implement signaling branching, e.g., for parallel procedure execution via concurrent propagation of state information to different NF groups (authentication, authorization, etc.), or optimization of NDB state updates (traffic sent to NF and NDB). HSDCN also releases NFs from the requirement to store the information about NF subscribers (stored in CBRR), thus reducing the coupling of NF instances. SFC via content-based routing can be effectively used in dynamic network topologies, such as Non-Terrestrial Network (NTN), as the network paths can be set up and modified in advance before changes to the network topology occur. This enables proactive and minimal NF reconfiguration, reducing costly CP exchanges.

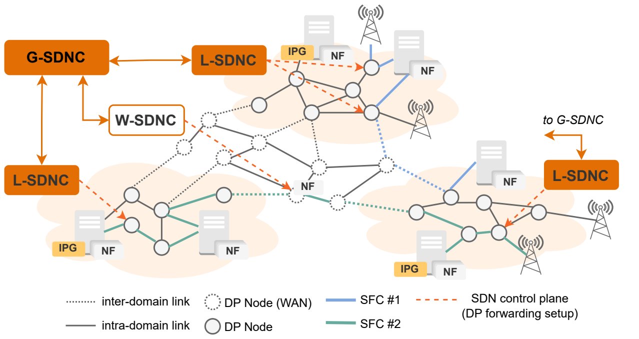


Fig. 4: Overview of SFC in distributed SDN network

IV. Evaluation

To maintain scalability, the operators usually deploy the core network with SCP as an NFs mediator (i.e., a star topology) instead of mesh deployments to offload NRF [5]. Hereby, we conduct a simplified simulation scenario to evaluate the impact of propagation delay between the NF of SCP-based and HSDCN deployment options. We consider two metrics: total transmission latency τ_t (being a sum of delays per procedure step τ_n) and total number of packet hops in the NF chain (corresponding to the consumed bandwidth):

$$\tau_t = \sum_{n}^{N} \tau_n \tag{1}$$

$$p_t = \sum_{n}^{N} |p_n| \tag{2}$$

where, p_n is the length of a path in hops between two NFs. For SCP variant (τ^{SCP}) each message between two NFs is transferred via SCP node w_{SCP}, whereas HSDCN (τ^{HSDCN}) sends the message directly to the NF. For NFs u and v the packet delay can be expressed as:

$$\tau_n^{SCP}(u, v) = \tau(u, w_{SCP}) + \tau(w_{SCP}, v) + \tau_p \tag{3}$$

$$\tau_n^{HSDCN} = \tau(u, v) + \tau_0 + \tau_p \tag{4}$$

, where τ_p – processing delay at switch (queuing, header matching, etc.), and τ_0 – overhead specific to CBRP induced by packet parsing and injection of tags into the packet headers.

To estimate the potential performance gains, we generated 2,500 random graphs (fat tree-like topologies) with node numbers varying from 20 to 50 nodes, corresponding to cloud, regional, and edge compute nodes, distributed uniformly over areas spanning a radius of 50 to 250 km. For link latency computation, we assumed propagation speed as in fiber (refractive $\eta \approx 1.44$). Then, we perform NF placement (5–10 NFs) over the graph nodes, while minimizing the latency between NFs. Finally, we define NF chains composed of 5 to 20 NFs that correspond to the execution of a network procedure. Fig. 5 shows the outcome of total latency and hop count for SCP-based and HSDCN deployments. We assume $\tau_0 \leq 1$ ms.

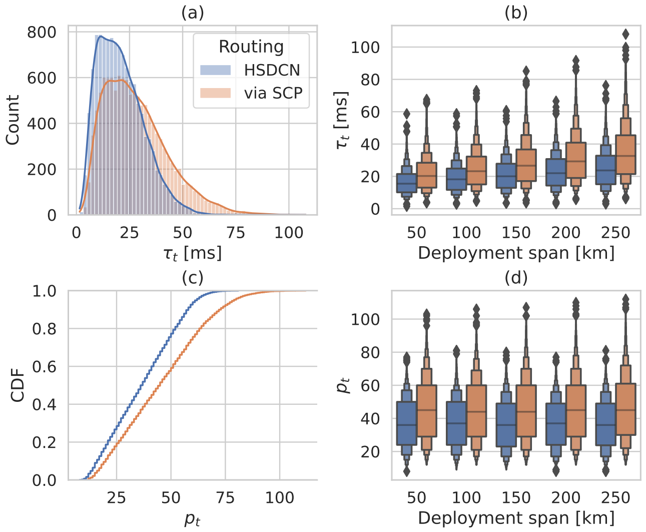


Fig. 5: Performance gains of HSDCN vs. SCP-based deployment in terms of: latency (a) and (b), and path length (c) and (d) for NFs exchange; (b) and (d) show the differences with reference to the deployment area size.

First, HSDCN enables a significant reduction of delay and bandwidth. On average, a 26% decrease in latency τ_t and 19% decrease of total path length p_t (corresponding to notable bandwidth savings) can be observed for the E2E procedures execution. The gain increases with the distance between the compute nodes on which the NFs are deployed, due to the higher propagation delay induced by centralized SCP routing. Moreover, the inter-quartile range for both τ_t and p_t is smaller for HSDCN, indicating suitability for time-sensitive use-cases, e.g., satellite-based ones. Moreover, it is worth noting that by using HSDCN, traffic concentration points can be easily avoided, which is problematic for SCP-based deployments, as it requires deploying additional SCP instances and further disaggregation of the network, thereby increasing related overheads. The above benefits, however, come at the cost of CBRP configuration and HSDCN framework deployment footprint, which require further investigation.

V. Summary and Conclusions

This paper proposes a hybrid CN architecture, HSDCN, bridging the gap between 5G and emerging 6G concepts. HSDCN combines 3GPP CN, a data-centric paradigm, and hierarchical multi-domain SDN to support incremental CN evolution. The framework introduces content-based and multi-path routing for CP traffic handling. The mechanisms enhance CN flexibility, enable dynamic complex SFC, facilitate signaling branching, and enable optimization of exchange between CN NFs. HSDCN also constitutes the scalable alternative for mesh CN deployments. Simplified simulations show 26% lower latency with content-based routing versus SCP-based CNs. Future work will focus on security extensions, implementation in the emulation environment, and evaluation of CP performance compared to 3GPP deployment models.

References

[1] NGMN, "6G requirements and design considerations," Next Generation Mobile Networks, White Paper ver. 1.0, Feb. 2023.
[2] Orange, "Mobile Network Technology Evolutions Beyond 2030," Orange, White Paper, Apr. 2024.
[3] M. Ericson, M. Condoluci, P. Rugeland, S. Wänstedt, M. S. Abad, O. Haliloglu, M. Saimler, and L. Feltrin, "6G architectural trends and enablers," in *2021 IEEE 4th 5G World Forum (5GWF)*, 2021, pp. 406–411.
[4] Y. Xing, J. Zhou, P. Li, and M. Yu, "Research on Key Technologies of 6G Network Architecture," in *2024 International Wireless Communications and Mobile Computing (IWCMC)*. Ayia Napa, Cyprus: IEEE, May 2024, pp. 451–455.
[5] G. Baldoni, J. Quevedo, C. Guimarães, A. de la Oliva, and A. Corsaro, "Data-Centric Service-Based Architecture for Edge-Native 6G Network," *IEEE Communications Magazine*, vol. 62, no. 4, pp. 32–38, Apr. 2024.
[6] G. Gur, A. Kalla, C. De Alwis, Q.-V. Pham, K.-H. Ngo, M. Liyanage, and P. Porambage, "Integration of ICN and MEC in 5G and Beyond Networks: Mutual Benefits, Use Cases, Challenges, Standardization, and Future Research," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1382–1412, 2022.
[7] U. Kulkarni, A. Sheoran, and S. Fahmy, "The Cost of Stateless Network Functions in 5G," in *Proceedings of the Symposium on Architectures for Networking and Communications Systems*. Layfette IN USA: ACM, Dec. 2021, pp. 73–79.
[8] C. Wernecke, H. Parzyjegla, and G. Mühl, "Implementing Content-based Publish/Subscribe on the Network Layer with P4," in *2020 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Nov. 2020, pp. 144–149.
[9] 3GPP, "System architecture for the 5G system (5GS)," 3rd Generation Partnership Project, Technical Standard TS 23.501, ver. 19.2.1, Jan. 2025.
[10] ——, "Procedures for the 5G system (5GS)," 3rd Generation Partnership Project, Technical Standard TS 23.502, ver. 18.5.0, Mar. 2024.
[11] ITU-R, "Future technology trends of terrestrial International Mobile Telecommunications systems towards 2030 and beyond," International Telecommunication Union – Radiocommunication Sector, Report M.2516-0, Nov. 2022.
[12] R. Liebhart, M. Shafi, H. Tataria, G. Shivanandan, and D. Chandramouli, "Perspectives on 6G Architectures," *IEEE Wireless Communications*, vol. 32, no. 1, pp. 108–114, Feb. 2025.
[13] X. Yan, X. An, W. Ye, M. Zhao, Y. Xi, and J. Wu, "User-Centric Network Architecture Design for 6G Mobile Communication Systems," in *2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. Gothenburg, Sweden: IEEE, Jun. 2023, pp. 305–310.
[14] S. Sthawarmath, E. Renault, and T. Lejkin, "Designing stateless control-plane for next-generation telecom networks," in *2022 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, 2022, pp. 154–159.
[15] M. Corici, F. Eichhorn, H. Buhr, and T. Magedanz, "Organic 6G networks: Ultra-flexibility through extensive stateless functional split," in *2023 2nd International Conference on 6G Networking (6gnet)*, 2023, pp. 1–8.
[16] E. Municio, G. García-Avilés, A. Bazco-Nogueras, M. Fiore, A. Collet, S. Alcalá, M. Camelo, P. Soto, D. D. Vleeschauwer, A. García-Saavedra, J. X. Salvat, A. Lutu, M. Gramaglia, A. Branchs, V. Prodomo, A. García, L. Fuentes, J. Ballesteros, M. Amor, M. Pinto, E. Kosmatos, and I. Páez, "Deliverable 2.3: Final DAEMON Network Intelligence framework and toolsets," Apr. 2024.
[17] H. Hantouti, N. Benamar, and T. Taleb, "Service Function Chaining in 5G & Beyond Networks: Challenges and Open Research Issues," Jan. 2022.
[18] M. Mihaeljans, A. Skrastins, and J. Porins, "Service Function Chaining via SR-MPLS Over P4: A Rudimentary Analysis," in *2024 International Conference on Intelligent Computing, Communication, Networking and Services (ICCNS)*. Dubrovnik, Croatia: IEEE, Sep. 2024, pp. 46–53.
[19] S. Bhowmik, "Content-Based Routing in Software-Defined Networks," Ph.D. dissertation, University of Stuttgart, 2018.
[20] Z. Wang, L. Wang, X. Gao, Y. Xia, and S. Wang, "An Architecture of Content-Centric Networking over Protocol-Oblivious Forwarding," in *2015 IEEE Globecom Workshops (GC Wkshps)*, Dec. 2015, pp. 1–5.

2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN) | 978-1-6654-3983-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/NFV-SDN53031.2021.9665052

Performance evaluation of the OSM orchestrator

Robert Kołakowski*◇, Lechosław Tomaszewski*, Sławomir Kukliński*◇
*Orange Polska, ◇Warsaw University of Technology
Warsaw, Poland

***Abstract*—The transformation of telecommunication networks towards fully softwarized solutions is ongoing. It requires efficient orchestration and management of softwarized networking solutions, including network slices. A widely accepted orchestration framework is already standardized by ETSI NFV, however, it is not yet validated by a commercial deployment and raises scalability concerns due to its highly centralized nature. The paper presents the results of experiments carried for performance evaluation of OSM, which is the most popular ETSI-compliant orchestration solution. The performance assessment is focused on the lifecycle operations on multiple instances of the Linux Foundation MAGMA vEPC template that are treated as network slices. The experiments have shown excellent scalability of the OSM orchestrator.**

***Index Terms*—network slicing, orchestration, NFV, OSM, performance evaluation, management, KPI, scalability**

I. Introduction

The 5G network had brought significant disruption to the telecommunication ecosystem. The concept uses the network virtualization concept with the ability to create several parallel and isolated networks satisfying specific service demands, i.e. network slices [1]. It is expected that the management and orchestration system of the 5G network will have to handle a vast number of slices with diverse lifecycle and runtime requirements, thus making effective management a crucial issue. Apart from the effective resource allocation to the network slices and satisfying the users' demands, the performance and scalability of the orchestration, which is responsible for the deployment, modification and termination of network slices, is of prime importance to slice tenants and network operators. A simple measure of the lifecycle operations performance can be slice deployment and termination time. Moreover, the behavior of the system can be different under the light and heavy load of the orchestration platform. Slice deployment time can be critical in short-lived slices and the ones that are created to handle emergency situations. Long slice termination time or slow resource allocation to the slices' components can lead to inefficient usage of resources and may degrade the quality of services of solutions that are deployed as a slice.

This paper is an experimental study focused on the performance evaluation of lifecycle-related procedures of an orchestrator compliant to European Telecommunications Standards Institute (ETSI) Network Functions Virtualisation (NFV) Management and Orchestration (MANO) framework. The performance assessments have been done using the ETSI Open Source Management and Orchestration (OSM) solution and relatively complex Network Slice Template (NST), i.e. Magma virtualized Evolved Packet Core (vEPC) developed by the Linux Foundation [2]. To assess the scalability of orchestration, the slice lifecycle KPIs proposed in [3] with an additional extension have been used.

This work has been supported by the EU H2020 projects: 5G-DRIVE (Grant Agreement No. 814956) and MonB5G (Grant Agreement No. 871780).

II. Problem statement and related work

The PM process is one of the fundamental activities of Network Operators. It serves both for Quality of Service (QoS) assurance and operators' environment expansion trends evaluation for proper anticipation of future demands to be responded in line with QoS targets. The hierarchy of the performance concerns reflects the hierarchy of service and network architecture as well as the underlying technologies. The numerous (thousands or more) Performance Indicators (PIs), composed either by direct or further reprocessed (recalculated, normalized, referred to the time of observation, etc.) readouts of specific counters, provide a quantitative view of the individual operation of equipment, functions, sub-systems and systems. To provide the practical and representative observability of the overall, end-to-end behavior of the systems, PIs are aggregated to higher-level abstraction performance indicators, forming a short list of Key Performance Indicators (KPIs). They further contribute to the end-to-end communication service-level quality representation by Key Quality Indicators (KQIs).

The Performance Management (PM) is in the area of interest of various Standards Developing Organizations. A review of their approaches is provided in [3]. It has to be noted that KPIs are used for the deployed solutions, but the management- and orchestration-related KPIs are so far rarely used. To that end, the 3rd Generation Partnership Project (3GPP) focuses on performance measurements of specific network functions (PIs) [4] and 5G end-to-end KPIs [5]. Management and performance evaluation of orchestration has been initially ignored, but later, within the Release 3 of ETSI NFV, the issues of orchestration management have been included to several specifications. The currently developed Release 4 has not provided substantial updates regarding this matter, yet. In [6] the detailed specification of the interfaces designed for management of NFV MANO framework has been provided, including interfaces for fault, configuration, performance and security management. Moreover, several performance metrics have been proposed (some being inline with the metrics already defined in [7]) that include physical resource usage such as CPU, RAM or storage, statistics regarding the objects managed by MANO or metrics related to lifecycle workflows (successful completions, failures, rollbacks etc.). These metrics can be exposed to external entities for further processing via the PM interface

978-1-6654-3983-1/21/$31.00 ©2021 IEEE

defined in details in [6]. The orchestration of slices can also be controlled by the Priority parameter of the Network Service Deployment Flavor [8]. The 5G System uses ETSI NFV MANO framework [9], however, 3GPP has extended the Operations/Business Support System (OSS/BSS) part of the ETSI NFV architecture by the components specific for orchestration and runtime management of 5G network slices. Furthermore, some functions associated with slice selection and authentication have also been added to the control plane of the 5G core.

The performance of network slices' orchestration processes is crucial for slice provisioning in a timeframe accepted by verticals (slice tenants) and for efficient utilization of resources during slice lifetime, which is also a key feature in the context of energy savings. Long slice termination or deployment time contributes to inefficient usage of resources and can cause excessive energy loss. It has to be noted that at the moment of writing the paper, there is none commercially deployed 5G network supporting network slicing. Therefore, the performance of the orchestration of network slicing has not been yet validated in large scale deployments.

The orchestration KPIs are so far ignored by 3GPP. In [3] the set of network slicing-related KPIs that are agnostic to orchestrated slices has been proposed. They are divided into two categories: real-time KPIs, dedicated to indicate over- and underutilization of slice resources, and slice lifecycle-oriented ones to evaluate the end-to-end operational agility of the network slicing platform. The lifecycle part of the approach has been used to evaluate the OSM [10] orchestrator, with the generic CirrosOS VNFs [11], [12]. The scalability of the slice instantiation process has also been evaluated in the paper. It has to be noted that the mentioned KPIs are not absolute measures as the execution time of slice lifecycle operations depends on many factors, especially the number of VNFs of slice template, their footprint, the number of parameters and procedures to be triggered in slice initialization phase or on the configuration and the performance of the infrastructure.

Some metrics for evaluation of the orchestration solutions as well as benchmarks for OSM also have already been described (cf. [13], [14]). The two types of KPIs have been proposed functional (i.a. resource footprint, the maximum number of supported objects) and operational KPIs (i.a. on-boarding process delay, deployment process delay or a metric for a total quantification of MANO lifecycle management performance called Quality of Decision). Moreover, the performance evaluation of OSM Release 4 has been done and compared with the Open Network Automation Platform (ONAP) orchestration solution.

III. Methodology

A. Definition of the measured KPIs

The list of the measured lifecycle KPIs proposed in [3] includes:

- Slice Deployment Time (SDT) – a parameter that describes the interval between slice deployment request and the moment when the slice is ready for operation. It depends on the performance of the orchestrator and the time needed for the allocation of resources by the virtualized infrastructure. Both times are dependent on the slice template complexity, which may deal with the footprint size of Virtual Network Functions (VNFs), their inter-connection topology, and the number of configuration parameters. Therefore, the required value of SDT cannot be defined in a generic case.
- Slice Deployment Time Scalability (SDTS) – a measure of scalability of slice deployment operations. To evaluate the scalability, it is possible to send *N* slice deployment requests of the same slice template and calculate SDTS as stated by the equation (1):

$$SDTS = \frac{GSDT}{N \cdot SDT} \tag{1}$$

  where *GSDT* is the overall time needed for the deployment of *N* identical slices and *SDT* is the deployment time of a single slice (as defined above).
- Slice Termination Time (STT) is a parameter that describes the interval between the slice termination request and the moment in which all slice allocated resources are released. If the time is long, it decreases the efficiency of the infrastructure resources usage (the resources are no more consumed).

The list is hereby extended with the newly defined KPI:

- Slice Termination Time Scalability (STTS) is a measure of the scalability of slice termination operations. The calculation of STTS, analogically to SDTS, is defined by the equation (2):

$$STTS = \frac{GSTT}{N \cdot STT} \tag{2}$$

  where *GSTT* is the time needed for termination of *N* identical slices together.

Both scalability KPIs, when lower than 1, show scalability gain on slice deployment/termination and resources allocation/deallocation, respectively.

B. Measurements of KPIs

To calculate the KPIs defined above, it is proposed to monitor the interactions between OSS/BSS and NFV Orchestrator (NFVO) with the NFV-MANO architecture [15]. The lifecycle KPIs can be obtained through the monitoring of the exchange of information between OSS/BSS and NFVO: OSS/BSS is able to determine both the beginning (request) and the end (completion report) of the requested procedure. There are two possible ways of calculation: (i) based on events logging in the on-board OSS/BSS log – each event is logged with a timestamp, and correlated search of beginning/finishing event for a specific procedure is sufficient; (ii) the OSS/BSS–MANO communication interface will typically use the time-out mechanism and the time-out timer can be implemented – its value at the end of the procedure may be instantly passed to the PM engine of OSS/BSS.

For the tests, the first approach has been chosen. Moreover, the load of the infrastructure (CPU, RAM) during slices orchestration operations has been measured.

IV. Testbed Description

A. Network slice templates used for the deployment

During the preliminary evaluation of the lifecycle KPIs [12], simple VNFs (i.e. lightweight Linux distributions) were used. Hence, the obtained results might not be in line with those for more demanding tasks, i.e. orchestration of complex slices composed of multiple VNFs, and requiring a lot of resources.

As recently considerable effort had been put in providing compatibility of OSM and Magma [2] solution developed by the Linux Foundation project, we have decided to further investigate the OSM orchestration capabilities [16] by conducting measurements of the lifecycle KPIs for deployment and termination of the Magma vEPC network slices, i.e. by the orchestration of multiple Magma instances. Due to the size and complexity of the orchestrated Magma VNFs, the measurement results enable a relatively good assessment of the real-life OSM performance.

Magma is an open source software platform enabling the creation of the cloudified 3GPP core networks (2G–5G) for different access networks. The solution is composed of the following three main components:

- Access Gateway (AGW) – providing network services and policy enforcement (in case of LTE network, AGW implements a single vEPC);
- Orchestrator (Orc8r) – cloud service to provide a way to configure and monitor the mobile network;
- Federation Gateway (FG) – a proxy to integrate AGWs and specific mobile core entities that contain subscribers' identities or charging policies (e.g. Home Subscriber Server, Policy and Charging Rules Function, etc.).

The interconnected AGW and orchestrator form the single Magma vEPC network slice, as illustrated in Fig. 1. FG is

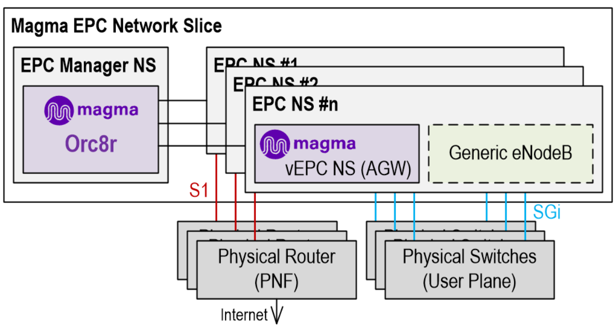


Fig. 1. High-level view of Magma EPC Network Slice (with multiple vEPC Network Slices)

an optional component outside the Magma vEPC network slice template. Moreover, the number of FGs would also be largely dependent on the spatial distribution of the core network entities (one gateway can proxy for multiple instances of Magma vEPC slices). Therefore, the measurements have been performed only for vEPC network slice (AGW).

B. Environment

The measurements of network slicing lifecycle KPIs have been conducted in the environment depicted in Fig. 2.

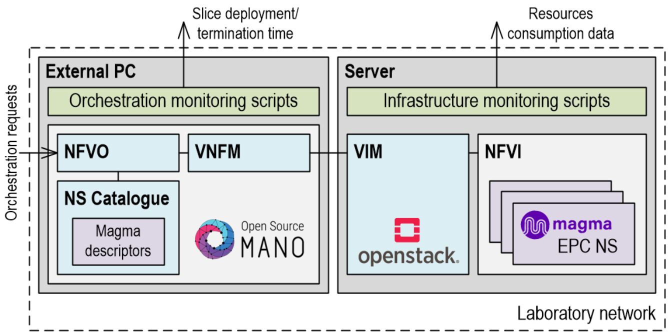


Fig. 2. The configuration of the measurement environment.

The test environment was composed of two machines: a PC hosting the OSM Release 9.1 and the server acting as Virtualized Infrastructure Manager (VIM), based on OpenStack, as well as NFV Infrastructure (NFVI) for the deployment of vEPC Network Slices. The parameters of the used hardware are presented in Table I.

TABLE I
SPECIFICATION OF THE HARDWARE USED DURING MEASUREMENTS

Hardware component	CPU	Memory	Disk
External PC	Intel® Core™ i7-8700 CPU @ 3.2 GHz	16 GB	256 GB
Server	Intel Xeon Silver 4216 2.1 GHz, 16C/32T, 9.6 GT/s	32 GB RDIMM 3200 MT/s	1 TB

The monitoring scripts were used on both machines during the orchestration operations. The scripts enabled the collection and extraction of the timestamps from OSM logs (PC case) and data related to resources consumption, i.e. RAM and CPU utilization (server case). The orchestration requests were generated by scripts using OSM Command Line Interface commands. Prior to testing the deployment and termination of vEPC Network Slices, one Magma Orc8r instance has been instantiated in NFVI (not shown in Fig. 2). To deploy the vEPC Network Slice Instances (NSIs), the following packages have been used:

- NS package: *magma-agw_nsd*
- VNF package: *magma-agw_vnfd*
- Image: *magma101.qcow2*

Each vEPC NSI required 2 GB of RAM, 50 GB of storage and 2 CPU cores to operate. Due to the hardware resource limitations, the largest possible number of concurrently running slice instances was 18 (due to lower slices' requirements for their operation without load). Therefore, the conducted KPI measurements were performed by deployment and termination from 1 to 18 identical instances using the above-mentioned template (translated first into the required format [17]). To assess the scalability KPIs, the operations have been done both in a simultaneous (bundled) way, as well as a one-by-one approach (deploying each additional instance once the previous deployment is finished).

To minimize the impact of random and undesired fluctuations of parameters during measurements, each test has been conducted ten times, and the obtained results have been averaged. Prior to performing the measurement tasks, the RAM and CPU usage in the idle state of the system has been measured and averaged. The obtained result was later on subtracted from the measurement results obtained during the orchestration tasks to separate the impact of orchestration processes on the resource consumption.

V. Results

A. Network slice deployment time measurements

The slice deployment process has been measured by deploying 1, 2, 3, 5, 7, 10, 14 and 18 AGW instances simultaneously (a bundle) and one-by-one, using the same template. Based on the measured SDT and GSDT values, the respective SDTS values for N given above have been calculated. The obtained results are presented in Fig. 3 and Fig. 4. The results of the AGW deployment measurements (cf. Fig. 3) show that the NSIs deployment bundling approach gives a clear advantage in comparison to the subsequent NSI deployment approach. The SDTS curve shows that the maximum gain on deployment bundling is achieved at a relatively small bundle size ($N = 5$), but for other N the SDTS values are still smaller than one.

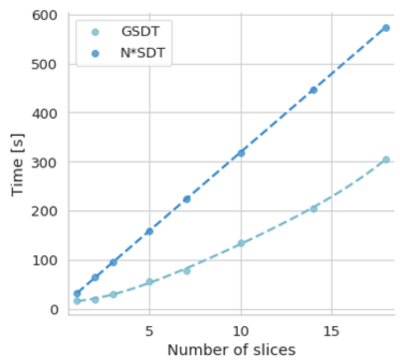


Fig. 3. Measured GSDT and N×SDT (for N = 1, 2, 3, 5, 7, 10, 14, 18).

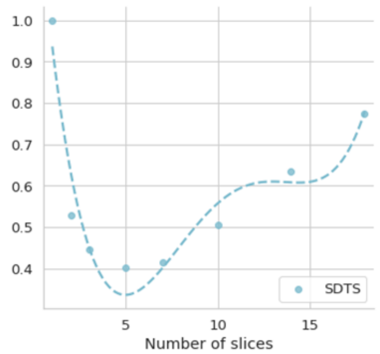


Fig. 4. Calculated SDTS (for N = 1, 2, 3, 5, 7, 10, 14, 18).

The averaged RAM and CPU consumption for a different number of slice instances during the slice deployment phase are presented in Fig. 5 and Fig. 6. These curves show trends that explain the behavior of the SDTS parameter and also show the load characteristics of the orchestration process. The deployment execution is preceded by a delay period, approximately independent of the bundle size, during which the resource orchestration processes prepare the environment (cf. Fig. 5). Then, the NSIs deployment process begins, which is manifested by a monotonic increase in memory usage without sudden and significant changes. However, in the same period of the deployment execution preparation, the peak of the CPU load, which is associated with increased VIM computation activity hosted on the same server (cf. Fig. 6) can be observed. Later, the load drops to the level resulting from the computational demand of the running NSIs.

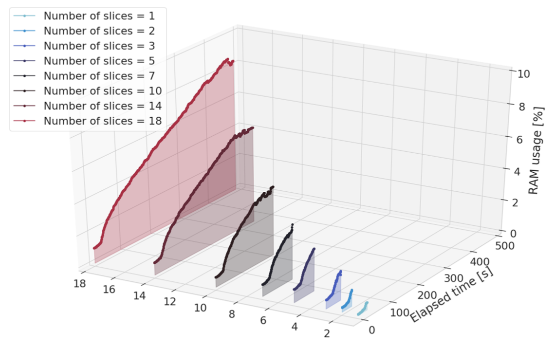


Fig. 5. Comparison of RAM usage measurement during the bundled deployment (for N = 1, 2, 3, 5, 7, 10, 14, 18).

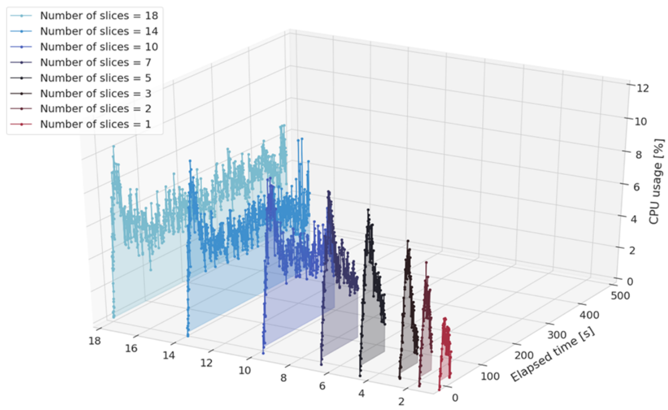


Fig. 6. Comparison of CPU usage measurement during the bundled deployment (for N = 1, 2, 3, 5, 7, 10, 14, 18).

B. Network slice termination time measurements

The slice termination measurements have been conducted for termination of 1, 2, 3, 5, 7, 10, 14 and 18 AGW instances in a simultaneous and one-by-one manner. It should be noted that during the series of measurements for $N = 18$, there was one termination process (out of 10), during which GSTT was much longer than in other cases and its influence is evident on all measurement curves.

Based on the measured STT and GSTT values, the respective STTS values for N given above have been calculated. The obtained results are presented in Fig. 7 and Fig. 8. Similar to the deployment processes, the NSIs termination bundling approach gives a time gain compared to the subsequent deployment approach. The gain, however, is not as big as in the case of deployment. Only for $N = 18$, the intersection of GSTT and N×STT curves is visible, but it is alleged (cf. Fig. 7), due to the mentioned disturbance in the mean time caused by one outstanding value in the termination process time. The maximum gain on termination bundling is here achieved at a very small bundle size ($N = 2$–3, cf. Fig. 8), but above this point, it is still time-efficient to combine deployment requests.

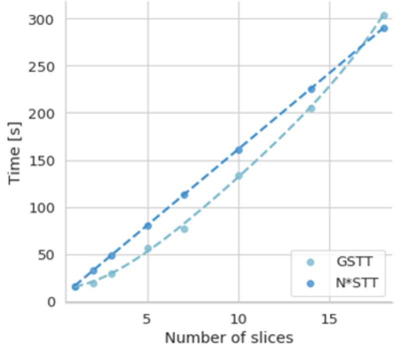


Fig. 7. Measured GSTT and N×STT (for $N = 1, 2, 3, 5, 7, 10, 14, 18$).

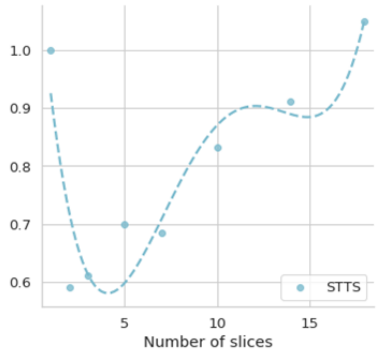


Fig. 8. Calculated STTS (for $N = 1, 2, 3, 5, 7, 10, 14, 18$).

The averaged results of RAM and CPU usage during the termination phase are presented in Fig. 9 and Fig. 10. The memory usage curves (cf. Fig. 9) show a monotonic decrease (note: for $N = 18$, the sudden jump up towards the end is related to the aforementioned disturbance in the average memory usage by one process out of 10, significantly longer than the others; the data series for the others ended earlier). For CPU usage curves during termination, the characteristic is a spike at the end of the process for all N values. Respecting the timescales, the curves for a specific N of both deployment and termination processes are approximately mutual mirror images. It has to be noted that the NSI termination time is, on average, only 50% shorter than the NSI deployment time.

C. Discussion on results

The comparison of both, deployment and termination processes performed simultaneously (bundled) and one-by-one approach, in the most demanding case ($N = 18$), is shown in Fig. 11 and Fig. 12 respectively. In this case, the termination process takes approximately the same time as the deployment time, whereas there is still a significant time gain for bundled

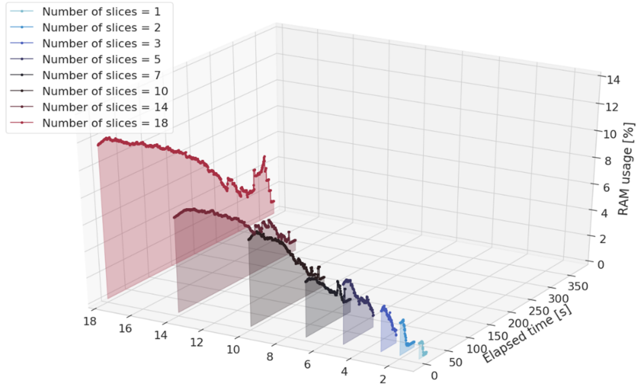


Fig. 9. Comparison of RAM usage measurement during the bundled termination (for $N = 1, 2, 3, 5, 7, 10, 14, 18$).

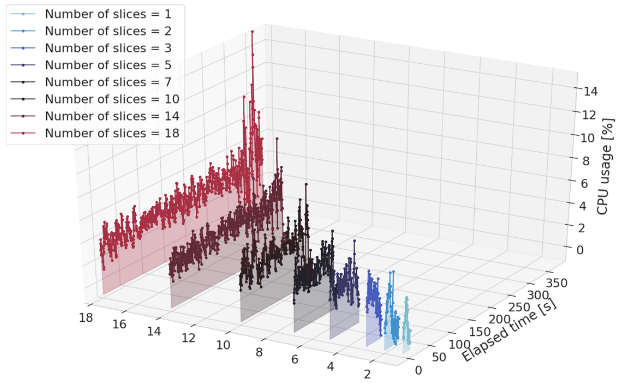


Fig. 10. Comparison of CPU usage measurement during the bundled termination (for $N = 1, 2, 3, 5, 7, 10, 14, 18$).

deployment (cf. Fig. 11). Considering the demand for resources during both processes, there is no significant difference in CPU utilization. However, the one-by-one approach is clearly more memory-eating (cf. Fig. 12).

It should be noted that the overall time of both kind processes in the one-by-one approach was calculated excluding the intervals between confirmations of the NSI deployment and sending the subsequent deployment request (only the request-confirmation intervals were considered). However, under the test conditions, the delay caused by OSM and its communication with VIM (order of ms) was negligible for the timescale of the entire process. The VIM-level processes had an overwhelming effect, but verification of the total dynamics should also be performed for OSM under stress conditions.

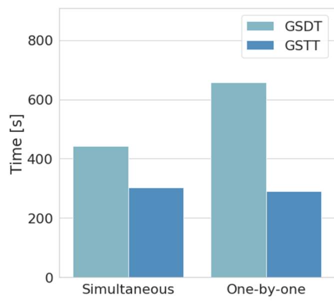


Fig. 11. Comparison of the overall time needed for simultaneous and one-by-one deployment and termination, $N = 18$.

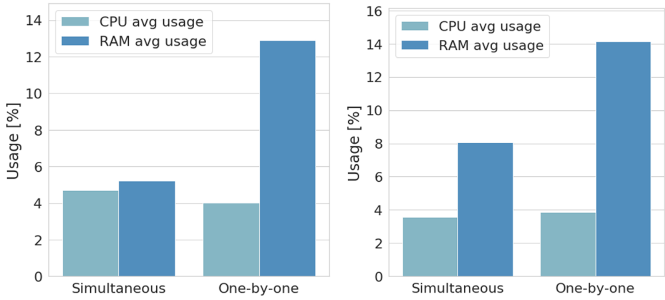


Fig. 12. Average values of CPU and RAM usage for both approaches to deployment (left) and termination (right), N = 18.

Comparing the results for Magma vEPC with those obtained for lightweight Virtual Machines (VMs) [12], one can see that in the case of the latter ones, the SDTS curve had a knee shape, falling to the level of 0.2 and stabilizing at this level (slight increase to the right), while for the current measurements the SDTS curve has an elbow shape, showing a significant increase to the right at much lower values of N. There are two major differences between the environments: (i) 12-thread CPU previously versus 32-thread CPU currently (however, supporting respectively 50 and 18 NSIs), and (ii) one-layer virtualization previously (NFV only) versus two-layer one currently (NFV + containerization). During the trials, the underlying inter-layer approach to the deployment of the worker nodes (VMs with Docker container runtime) for implementation of containers (embedding the network functions) was left unchanged. The impact at this level of either one-by-one strategy or bundled strategy on the orchestration performance should be further studied and validated.

VI. Summary and Conclusion

The MANO orchestration is a complex process composed of multiple sub-processes. Its implementation is even more complicated as it has to deal with the deployed platform and operating system internal processes (garbage collection, etc.) that can be scheduled randomly by the operating system. Therefore, the full analytical approach to the orchestration process is problematic. It is, however, possible to make some assessment of the orchestration performance using experiments. This is the way which we have followed. The results of the experiments that we have done for the performance evaluation of OSM, the ETSI-compliant MANO orchestrator, are a bit surprising. Overall, the OSM orchestrator has proved to achieve very decent performance. The duration of a slice instantiation or termination phases are not negligible but at a satisfactory level. It is surprising that in the experiments, the slice termination time was relatively high and, in all cases, was equal to or exceeded 50% of slice deployment time. The deployment of a single Magma vEPC slice took $\sim$30 s, which is surprisingly fast. On the other hand, the typical termination time for a single instance, which took $\sim$20 s, is much longer than our expectations. With the increasing number of slices, the slice deployment and termination times increased less than linearly, showing OSM's excellent scalability. The "bundled mode" of slice orchestration has outperformed the one-by-one approach. In real life, however, such an approach will be hardly achievable – the requests are sent randomly and the scheduling of slice deployment or termination requests can be used only in large networks. Nonetheless, orchestration scheduling can be beneficial for network slice calendaring i.e. deploying or terminating a slice for the predetermined date.

The non-monotonicity of SDTS and STTS curves cannot be explained without the performance analysis of the orchestration platform's internal blocks. Such an approach is hardly achievable but is needed to find the performance bottleneck.

In future works, we plan to obtain more results for different slice templates and infrastructure configurations (more servers). Moreover, some experiments concerning the scalability of resource scaling (during slice runtime), omitted in the paper, are planned to be done.

Acknowledgment

The authors thank Michał Wierzbicki, Łukasz Maksimowski and Michał Rosiński for their significant contributions to the preparation of the test environment and to the measurements.

References

[1] NGMN Alliance, "NGMN 5G White Paper", Jan. 2015.

[2] Magma, "An Open-Source Platform For Building Carrier-Grade Networks", [Online]. Available: https://www.magmacore.org/ (accessed Oct. 10, 2021).

[3] S. Kukliński and L. Tomaszewski, "Key Performance Indicators for 5G network slicing", 2019 IEEE Conference on Network Softwarization (NetSoft), 2019, pp. 464-471, doi:10.1109/NETSOFT.2019.8806692.

[4] 3GPP, "Management and orchestration; 5G performance measurements", 3GPP TS 28.552, ver. 17.4.0, Sep. 2021.

[5] 3GPP, "Management and orchestration; 5G end to end Key Performance Indicators (KPI)", 3GPP TS 28.554, ver. 17.4.0, Sep. 2021.

[6] ETSI, "Requirements and interfaces specification for management of NFV-MANO", ETSI GS NFV-IFA 031, V4.2.1, Jun. 2021.

[7] ETSI, "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Performance Measurements Specification", ETSI GS NFV-IFA 027, V4.2.1, May 2021.

[8] ETSI, "Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Network Service Templates Specification", ETSI GS NFV-IFA 014, V4.2.1, May 2021.

[9] 3GPP, "Management of network slicing in mobile networks; Architecture framework", 3GPP TS 28.533, ver. 17.0.0, Sep. 2021.

[10] ETSI, "Open Source MANO", [Online]. Available: https://osm.etsi.org/ (accessed Oct. 10, 2021).

[11] ETSI, "OSM Release SIX – Deploying your first Network Service", [Online]. Available: https://osm.etsi.org/wikipub/index.php/OSM_Release_SIX (accessed Oct. 10, 2021).

[12] M. A. Kourtis et al., "5G Network Slicing Enabling Edge Services", 2020 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), 2020, pp. 155160, doi:10.1109/NFV-SDN50289.2020.9289880.

[13] G. M. Yilma, F. Z. Yousaf, V. Sciancalepore, X. Costa-Perez, "Benchmarking Open-Source NFV MANO Systems: OSM and ONAP", Computer Communications, vol. 161, pp. 86-98, ISSN 0140-3664, 2020. doi:10.1016/j.comcom.2020.07.013

[14] G. M. Yilma, F. Z. Yousaf, V. Sciancalepore, X. Costa-Perez, "On the Challenges and KPIs for Benchmarking Open-Source NFV MANO Systems: OSM vs ONAP.", 2019, [Online]. Available: https://arxiv.org/abs/1904.10697 (accessed Oct. 10, 2021)

[15] ETSI, "Network Functions Virtualisation (NFV); Management and Orchestration", ETSI GS NFV-MAN 001, V1.1.1, Dec. 2014.

[16] OSM, "OSM architecture and functions", [Online]. Available: https://osm.etsi.org/docs/user-guide/02-osm-architecture-and-functions.html (accessed Oct. 10, 2021).

[17] ETSI, "Network Functions Virtualisation (NFV) Release 3; Protocols and Data Models; NFV descriptors based on YANG Specification", ETSI GS NFV-SOL 006, V3.5.1, Jul. 2021.

Appendix B

Declarations of co-authors

This section comprises the declarations of co-authors and their contributions to the research works that are a part of the dissertation. The contributions are expressed by percentage and roles following the universal Contributor Role Taxonomy(CRediT) `https://credit.niso.org/`. The declarations are listed below:

[A1] Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Evaluating Scalability of Hierarchical Multi-controller SDN: A Holistic Approach". In: *Journal of Network and Systems Management* 34.4 (May 2026), p. 107. ISSN: 1573-7705. DOI: `10.1007/s10922-026-10087-w` – p. 220

[A2] Robert Kołakowski, Slawomir Kukliński, and Lechosław Tomaszewski. "Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Networks". In: *2024 IFIP Networking Conference (IFIP Networking)*. June 2024, pp. 607–612. DOI: `10.23919/IFIPNetworking62109.2024.10619899` – p. 221

[A3] Robert Kołakowski et al. "Hierarchical Traffic Engineering in 3D Networks Using QoS-Aware Graph-Based Deep Reinforcement Learning". In: *Electronics* 14.5 (Mar. 2025), p. 1045. ISSN: 2079-9292. DOI: `10.3390/electronics14051045` – p. 222

[A4] Robert Kołakowski, Sławomir Kukliński, and Lechosław Tomaszewski. "Time-of-Day-Aware Slice Admission Control". In: *2023 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*. Sept. 2023, pp. 199–204. DOI: `10.1109/MeditCom58224.2023.10266645` – p. 223

[A5] Lechosław Tomaszewski and Robert Kołakowski. "Advanced Air Mobility and Evolution of Mobile Networks". In: *Drones* 7.9 (Sept. 2023), p. 556. ISSN: 2504-446X. DOI: `10.3390/drones7090556`. (Visited on 02/27/2025) – p. 224

[A6] Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "On 6G-Enabled SDN-Based Mobile Network User Plane with DRL-Based Traffic Engineering". In: *Artificial Intelligence Applications and Innovations. AIAI 2024 IFIP WG 12.5 International Workshops*. Ed. by Ilias Maglogiannis et al. Cham: Springer Nature Switzerland, 2024, pp. 124–138. ISBN: 978-3-031-63227-3. DOI: `10.1007/978-3-031-63227-3_9` – p. 225

[A7] Robert Kołakowski, Thierry Lejkin, and Victorien Romain. "Hybrid SDN-Based Data-Centric Mobile Core Network with Content-Based Routing". In: *2025 International Conference on*

6G Networking (6GNet 2025). Paris, France, Dec. 2025, pp. 126–130. DOI: `10.1109/6GNet68413.2025.11314080` – p. 226

[A8] Robert Kołakowski, Lechosław Tomaszewski, and Sławomir Kukliński. "Performance evaluation of the OSM orchestrator". In: *2021 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. Nov. 2021, pp. 15–20. DOI: `10.1109/NFV-SDN53031.2021.9665052` – p. 227

Warsaw, 02.12.2025

Declaration of co-authors

The contributions of the individual co-authors to the preparation of the article **"Evaluating Scalability of Hierarchical Multi-Controller SDN: A Holistic Approach", DOI:** 10.1007/s10922-026-10087-w are hereby declared.

Co-author	Contribution [%]	Contributor Roles	Signature
Robert Kołakowski	80	Conceptualisation Data curation Formal Analysis Investigation Methodology Software Validation Visualisation Writing–original draft preparation Writing–review and editing	Robert Kołakowski
Sławomir Kukliński	15	Conceptualisation Formal Analysis Investigation Writing–original draft preparation Writing–review and editing Supervision	Sławomir Kukliński
Lechosław Tomaszewski	5	Formal Analysis Writing–original draft preparation Writing–review and editing	[illegible]

Warsaw, 02.12.2025

Declaration of co-authors

The contributions of the individual co-authors to the preparation of the article **"Hierarchical Deep Reinforcement Learning-Based Load Balancing Algorithm for Multi-Domain Software-Defined Networks", DOI: 10.23919/IFIPNetworking62109.2024.10619899** are hereby declared.

Co-author	Contribution [%]	Contributor Roles	Signature
Robert Kołakowski	75	Conceptualisation Data Curation Formal Analysis Investigation Methodology Software Validation Visualisation Writing–original draft preparation Writing–review and editing	Robert Kołakowski
Sławomir Kukliński	20	Conceptualisation Formal Analysis Investigation Methodology Supervision Writing–original draft preparation Writing–review and editing	Sławomir Kukliński
Lechosław Tomaszewski	5	Writing–original draft preparation Writing–review and editing	[illegible]

Warsaw, 02.12.2025

Declaration of co-authors

The contributions of the individual co-authors to the preparation of the article **"Hierarchical Traffic Engineering in 3D Networks Using QoS-aware Graph-based Deep Reinforcement Learning", DOI: 10.3390/electronics14051045** are hereby declared.

Co-author	Contribution [%]	Contributor Roles	Signature
Robert Kołakowski	75	Conceptualisation Data Curation Formal Analysis Investigation Methodology Software Validation Visualisation Writing–original draft preparation Writing–review and editing	Robert Kołakowski
Lechosław Tomaszewski	10	Writing–original draft preparation Writing–review and editing	[illegible]
Rafał Tępiński	5	Conceptualisation Software Writing–original draft preparation	Rafał Tępiński
Sławomir Kukliński	10	Writing–original draft preparation Writing–review and editing	Sławomir Kukliński

Warsaw, 02.12.2025

Declaration of co-authors

The contributions of the individual co-authors to the preparation of the article **"Time-of-Day-Aware Slice Admission Control", DOI: 10.1109/MeditCom58224.2023.10266645** are hereby declared.

Co-author	Contribution [%]	Contributor Roles	Signature
Robert Kołakowski	60	Conceptualisation Data curation Formal Analysis Investigation Methodology Software Validation Visualization Writing—original draft preparation Writing—review and editing Visualisation	
Sławomir Kukliński	30	Conceptualisation Investigation Methodology Writing—original draft preparation Writing—review and editing Visualisation	
Lechosław Tomaszewski	10	Investigation Writing—original draft preparation Writing—review and editing	

Warsaw, 02.12.2025

Declaration of co-authors

The contributions of the individual co-authors to the preparation of the article **"Advanced Air Mobility and Evolution of Mobile Networks", DOI: 10.3390/drones7090556** are hereby declared.

Co-author	Contribution [%]	Contributor Roles	Signature
Lechosław Tomaszewski	65	Conceptualisation Methodology Software Validation Formal Analysis Investigation Data Curation Writing–original draft preparation Writing–review and editing Visualisation	
Robert Kołakowski	35	Conceptualisation Methodology Software Validation Formal Analysis Investigation Data Curation Writing–original draft preparation Writing–review and editing Visualisation	

Warsaw, 02.12.2025

Declaration of co-authors

The contributions of the individual co-authors to the preparation of the article **"On 6G-Enabled SDN-Based Mobile Network User Plane with DRL-Based Traffic Engineering", DOI: 10.1007/978-3-031-63227-3_9** are hereby declared.

Co-author	Contribution [%]	Contributor Roles	Signature
Robert Kołakowski	80	Conceptualisation Investigation Writing–original draft preparation Writing–review and editing Visualisation	
Lechosław Tomaszewski	10	Writing–original draft preparation Writing–review and editing Visualisation	
Sławomir Kukliński	10	Formal Analysis Writing–original draft preparation Writing–review and editing Supervision	

Warsaw, 02.12.2025

Declaration of co-authors

The contributions of the individual co-authors to the preparation of the article **"Hybrid SDN-Based Data-Centric Mobile Core Network with Content-Based Routing", DOI:10.1109/6GNet68413.2025.11314080** are hereby declared.

Co-author	Contribution [%]	Contributor Roles	Signature
Robert Kołakowski	80	Conceptualisation Data curation Formal Analysis Investigation Methodology Software Validation Visualisation Writing—original draft preparation Writing—review and editing	Robert Kołakowski
Thierry Lejkin	10	Conceptualisation Writing—original draft preparation	—//—
Victorien Romain	10	Conceptualisation Writing—original draft preparation	—//—

Warsaw, 02.12.2025

Declaration of co-authors

The contributions of the individual co-authors to the preparation of the article **"Performance Evaluation of the OSM Orchestrator", DOI: 10.1109/NFV-SDN53031.2021.9665052** are hereby declared.

Co-author	Contribution [%]	Contributor Roles	Signature
Robert Kołakowski	55	Conceptualisation Software Formal Analysis Investigation Methodology Validation Writing–original draft preparation Writing–review and editing Visualisation	
Lechosław Tomaszewski	15	Conceptualisation Investigation Writing–original draft preparation Writing–review and editing	
Sławomir Kukliński	30	Conceptualisation Formal Analysis Investigation Methodology Writing–original draft preparation Writing–review and editing Supervision	