



*future internet*

# 5G Enabling Technologies and Wireless Networking

---

Edited by

Michael Mackay

Printed Edition of the Special Issue Published in *Future Internet*

# **5G Enabling Technologies and Wireless Networking**



# 5G Enabling Technologies and Wireless Networking

Editor

**Michael Mackay**

MDPI • Basel • Beijing • Wuhan • Barcelona • Belgrade • Manchester • Tokyo • Cluj • Tianjin



*Editor*

Michael Mackay

Liverpool John Moores University

UK

*Editorial Office*

MDPI

St. Alban-Anlage 66

4052 Basel, Switzerland

This is a reprint of articles from the Special Issue published online in the open access journal *Future Internet* (ISSN 1999-5903) (available at: [https://www.mdpi.com/journal/futureinternet/special\\_issues/5GET\\_WN](https://www.mdpi.com/journal/futureinternet/special_issues/5GET_WN)).

For citation purposes, cite each article independently as indicated on the article page online and as indicated below:

LastName, A.A.; LastName, B.B.; LastName, C.C. Article Title. <i>Journal Name</i> <b>Year</b> , <i>Volume Number</i> , Page Range.
--

**ISBN 978-3-0365-6806-5 (Hbk)**

**ISBN 978-3-0365-6807-2 (PDF)**

© 2023 by the authors. Articles in this book are Open Access and distributed under the Creative Commons Attribution (CC BY) license, which allows users to download, copy and build upon published articles, as long as the author and publisher are properly credited, which ensures maximum dissemination and a wider impact of our publications.

The book as a whole is distributed by MDPI under the terms and conditions of the Creative Commons license CC BY-NC-ND.

# Contents

<b>About the Editor</b> . . . . .	<b>vii</b>
<b>Preface to “5G Enabling Technologies and Wireless Networking”</b> . . . . .	<b>ix</b>
<b>Michael Mackay</b> Editorial for the Special Issue on 5G Enabling Technologies and Wireless Networking Reprinted from: <i>Future Internet</i> <b>2022</b> , <i>14</i> , 342, doi:10.3390/fi14110342 . . . . .	<b>1</b>
<b>Michael Mackay, Alessandro Raschella and Ogeen Toma</b> Modelling and Analysis of Performance Characteristics in a 60 Ghz 802.11ad Wireless Mesh Backhaul Network for an Urban 5G Deployment Reprinted from: <i>Future Internet</i> <b>2022</b> , <i>14</i> , 34, doi:10.3390/fi14020034 . . . . .	<b>3</b>
<b>Lorenzo Ricciardi Celsi, Andrea Caliciotti, Matteo D’Onorio, Eugenio Scocchi, Nour Alhuda Sulieman, Massimo Villari</b> On Predicting Ticket Reopening for Improving Customer Service in 5G Fiber Optic Networks Reprinted from: <i>Future Internet</i> <b>2021</b> , <i>13</i> , 259, doi:10.3390/coatings11030274 . . . . .	<b>19</b>
<b>Lopamudra Hota, Biraja Prasad Nayak, Arun Kumar, G. G. Md. Nawaz Ali and Peter H.J. Chong</b> An Analysis on Contemporary MAC Layer Protocols in Vehicular Networks: State-of-the-Art and Future Directions Reprinted from: <i>Future Internet</i> <b>2021</b> , <i>13</i> , 287, doi:10.3390/fi13110287 . . . . .	<b>35</b>
<b>Jesús Fernando Cevallos Moreno, Rebecca Sattler, Raúl P. Caulier Cisterna, Lorenzo Ricciardi Celsi, Aminael Sánchez Rodríguez and Massimo Mecella</b> Online Service Function Chain Deployment for Live-Streaming in Virtualized Content Delivery Networks: A Deep Reinforcement Learning Approach Reprinted from: <b>2021</b> , <i>13</i> , 278, doi:10.3390/fi13110278 . . . . .	<b>81</b>
<b>Andreas G. Papidas and George C. Polyzos</b> Self-Organizing Networks for 5G and Beyond: A View from the Top Reprinted from: <i>Future Internet</i> <b>2022</b> , <i>14</i> , 95, doi:10.3390/fi14030095 . . . . .	<b>109</b>



## About the Editor

### **Michael Mackay**

Michael Mackay (Dr) is a Senior Lecturer in the School of Computing and Mathematical Sciences at Liverpool John Moores University and is currently leading the Networking and Distributed Systems Group. He received his PhD in IPv6 Transitioning from Lancaster University in 2005. Dr Mackay has over 20 years of experience in network research, which covers a broad range of topics, including performance, quality of service, mobility, and cloud/edge systems. His current focus is on heterogeneous 5G/6G wireless network management and edge convergence.



# Preface to “5G Enabling Technologies and Wireless Networking”

This Special Issue of *Future Internet* focuses on research related to the on-going deployment of 5G wireless networks and the new technologies that are being developed to underpin it. We cover a wide range of topics on this subject, including network performance and new management approaches, and our motivation is to lay a foundation for beyond 5G networks as we begin to consider 6G going forward. This collection is of interest to any researchers or operators who are considering 5G networks in a range of scopes, including traditional mobile network operators (MNOs), non-public networks (NPNs), and a range of use cases, including vehicular networks. We would like to thank all the authors who have contributed to this Special Issue and the editorial team at MDPI for their hard work in facilitating the publication of this Special Issue.

**Michael Mackay**  
*Editor*





Editorial

# Editorial for the Special Issue on 5G Enabling Technologies and Wireless Networking

Michael Mackay

School of Computer Science and Mathematics, Liverpool John Moores University, Byrom Street,  
Liverpool L3 3AF, UK; m.i.mackay@ljmu.ac.uk

The ongoing deployment of 5G networks is seen as a key enabler for realizing upcoming interconnected services at scale, including the massive deployment of the Internet of Things, providing V2X communications to support autonomous vehicles, and the increase in smart homes, smart cities, and Industry 4.0. The Special Issue of *Future Internet* titled “5G Enabling Technologies and Wireless Networking” focuses on 5G-enabling technologies to meet these access, efficiency, and performance requirements and facilitates coordinating wireless networks to support future deployments. We published a total of five papers that covered a range of topics from practical 5G deployment experiences in non-public and vehicular networks to applying technologies such as service chaining and self-organising Networks (SONs) to network operations. As such, we have grouped the published work into two broad categories: the first discusses practical issues related to 5G deployments while the second looks at enabling technologies and how they can supplement or extend current existing 5G networks.

In the first group, we have three papers focusing on a range of issues related to current 5G wireless networks. In the first paper [1], the authors present the results of their work to model and predict the performance of millimeter wave (mmWave) backhaul links that were deployed as part of the Liverpool 5G network. Based on the properties of the 802.11ad protocol and the physical characteristics of the environment, they simulated how each link performed with different signal-to-noise ratios (SNR) and packet error rate (PER) values and verified them against real-world deployed links. The results showed a good convergence between the simulated and real results and provide a solid foundation for further network planning and optimization. This type of practical deployment experience, particularly around non-public networks also provides useful insights into how 5G deployments can be optimised going forward. In the second paper, Ricciardi Celsi et al. [2] took a different approach and proposed a data-driven strategy for predicting customer service technical ticket reopening for 5G fiber telecommunications companies. Their main aim was to ensure that the service level agreement between the end user and service provider was satisfied in terms of the perceived quality of service. The authors made a detailed comparison of different approaches to classification—ranging from decision trees to artificial neural networks and support vector machines—and found that a Bayesian network classifier is the most accurate at predicting whether a monitored ticket will be reopened or not. This work again provides useful insights into how 5G networks can be managed as user numbers continue to grow and become denser and the potential for congestion or other issues increases. Finally, from a practical perspective, Hota et al. [3] reviewed the applications, characteristics, and challenges faced in the design of MAC protocols in 5G vehicular environments. They presented a classification of MAC protocols based on the metrics of contention mechanisms and channel access. In the first case, contention was listed as contention-based, contention-free, and hybrid, whereas channel access was categorized as being distributed, centralized, cluster-based, cooperative, token-based, or random access. The paper gives an analysis of the objectives, mechanisms, advantages/disadvantages, and simulators used in each protocol and provides a discussion on the future scope and

**Citation:** Mackay, M. Editorial for the Special Issue on 5G Enabling Technologies and Wireless Networking. *Future Internet* **2022**, *14*, 342. <https://doi.org/10.3390/fi14110342>

Received: 16 November 2022

Accepted: 17 November 2022

Published: 21 November 2022

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

open challenges for improving MAC protocol's design. This is important because vehicular networks continue to be an important use case for 5G so new insights into MAC protocol's design may lead to performance improvements in the future as these networks expand.

In the second group, we consider more theoretical supporting technologies that can be utilised to enhance the performance of existing 5G networks. The paper by Moreno et al. [4] presented a multi-objective optimization framework for service function chain deployment in the context of live-streaming in virtualized content delivery networks. Specifically, they developed an enhanced exploration, dense reward mechanism over a Dueling Double Deep Q Network (E2-D4QN) for container-based network function virtualization. Their simulation results demonstrated that their approach can provide substantial QoS/QoE performance improvements and adapt to the complexities of live-video deliveries for general-case service function chain deployments. NFV and service chaining are important concepts in networking research and are seen as fundamental concepts for future 5G networks. In addition to this, in Papidas and Polyzos [5], the authors described self-organizing network (SON) concepts and architectures and their potential to play a central role in 5G deployment, focusing on a basic SON use case applied to radio access networks (RANs). They first analyzed SON applications' rationale and operation and the design and dimensioning of SON systems before highlighting possible deficiencies and conflicts that occur via the parallel operation of functions. As part of this, they also described the strong reliance on machine learning (ML) and artificial intelligence (AI) to enable this approach. Finally, they presented and commented on recent proposals for SON deployments in 5G networks. As stated above, SON is seen as a very desirable feature in future wireless networks due to the dynamic nature of the medium and the need for continuous rapid adjustments to both maximise performances and to ensure robustness.

In conclusion, this Special Issue presented a range of papers that present an up-to-date view on the ongoing 5G roll-out and the issues that are currently being addressed in the research community. On the one hand, now that 5G is deployed, we are starting to gain important insights into how the current generation of wireless networks perform in practice and how they can be managed effectively or applied in different use cases. Meanwhile, on the other hand, we are now starting to explore how new technologies can be integrated into the 5G platform to lay the foundation for future generations as we move towards Beyond 5G and 6G networks.

Finally, we would like to thank all authors for their submitted papers to this Special Issue. We would also like to acknowledge all reviewers for their careful and timely reviews in helping to improve the quality of this Special Issue.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Mackay, M.; Raschella, A.; Toma, O. Modelling and Analysis of Performance Characteristics in a 60 Ghz 802.11ad Wireless Mesh Backhaul Network for an Urban 5G Deployment. *Future Internet* **2022**, *14*, 34. [[CrossRef](#)]
2. Ricciardi Celsi, L.; Caliciotti, A.; D'Onorio, M.; Scocchi, E.; Sulieman, N.A.; Villari, M. On Predicting Ticket Reopening for Improving Customer Service in 5G Fiber Optic Networks. *Future Internet* **2021**, *13*, 259. [[CrossRef](#)]
3. Hota, L.; Nayak, B.P.; Kumar, A.; Ali, G.G.M.N.; Chong, P.H.J. An Analysis on Contemporary MAC Layer Protocols in Vehicular Networks: State-of-the-Art and Future Directions. *Future Internet* **2021**, *13*, 287. [[CrossRef](#)]
4. Cevallos Moreno, J.F.; Sattler, R.; Caulier Cisterna, R.P.; Ricciardi Celsi, L.; Sánchez Rodríguez, A.; Mecella, M. Online Service Function Chain Deployment for Live-Streaming in Virtualized Content Delivery Networks: A Deep Reinforcement Learning Approach. *Future Internet* **2021**, *13*, 278. [[CrossRef](#)]
5. Papidas, A.G.; Polyzos, G.C. Self-Organizing Networks for 5G and Beyond: A View from the Top. *Future Internet* **2022**, *14*, 95. [[CrossRef](#)]



## Article

# Modelling and Analysis of Performance Characteristics in a 60 Ghz 802.11ad Wireless Mesh Backhaul Network for an Urban 5G Deployment

Michael Mackay \*, Alessandro Raschella and Ogeen Toma

School of Computer Science and Mathematics, Liverpool John Moores University, Byrom Street, Liverpool L3 3AF, UK; A.Raschella@ljmu.ac.uk (A.R.); O.Toma@ljmu.ac.uk (O.T.)

\* Correspondence: M.I.Mackay@ljmu.ac.uk

**Abstract:** With the widespread deployment of 5G gaining pace, there is increasing interest in deploying this technology beyond traditional Mobile Network Operators (MNO) into private and community scenarios. These deployments leverage the flexibility of 5G itself to support private networks that sit alongside or even on top of existing public 5G. By utilizing a range of virtualisation and slicing techniques in the 5G Core (5GC) and heterogeneous Radio Access Networks (RAN) at the edge, a wide variety of use cases can be supported by 5G. However, these non-typical deployments may experience different performance characteristics as they adapt to their specific scenario. In this paper we present the results of our work to model and predict the performance of millimeter wave (mmWave) backhaul links that were deployed as part of the Liverpool 5G network. Based on the properties of the 802.11ad protocol and the physical characteristics of the environment, we simulate how each link will perform with different signal-to-noise ratio (SNR) and Packet Error Rate (PER) values and verify them against real-world deployed links. Our results show good convergence between simulated and real results and provide a solid foundation for further network planning and optimization.

**Citation:** Mackay, M.; Raschella, A.; Toma, O. Modelling and Analysis of Performance Characteristics in a 60 Ghz 802.11ad Wireless Mesh Backhaul Network for an Urban 5G Deployment. *Future Internet* **2022**, *14*, 34. <https://doi.org/10.3390/fi14020034>

Academic Editor: Paolo Bellavista

Received: 17 December 2021

Accepted: 18 January 2022

Published: 21 January 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** private 5G; mmWave RAN; modulation and coding schemes; spectral efficiency

## 1. Introduction

The ongoing deployment of 5G networks marks the start of next evolution wireless networking as technologies converge and use cases extend beyond traditional home/business use. This has implications for a range of environments, from dense urban deployments right through to sparse rural usage. Such deployments will necessitate end-to-end and top-to-bottom flexibility in terms of the mix of Radio Access Network (RAN) technologies used and how traditional back-ends are deployed (or not). In particular, 5G deployments are now being considered by providers beyond traditional Mobile Network Operators (MNOs) to provide connectivity and services for a wide range of uses. These include private 5G for use in Industrial Internet of Things (IIoT) and Industry 4.0 use cases, and community-based efforts such as those delivered by Liverpool 5G.

Due to this growth, 'backhauling' has become a central challenge for operators in order to provide multi-gigabit capacity while using cost efficient technologies [1]. Backhaul solutions can be categorized as wired (leased lines or copper/fibre) or wireless (point-to-point, point-to-multipoint over high-capacity radio links). Wired solutions are typically an expensive solution but offer unlimited bandwidth and ease of maintenance [1]. On the other hand, wireless solutions have the advantage of rapid and easy deployment at relatively low cost. In mobile networks, backhauling is expected to be filled by the 5G NR FR1 and FR2 standards but may be limited to licensed operators and incur significant costs and overheads. Therefore, millimetre wave (mmWave) techniques ranging from 30 to 300 GHz have become a feasible alternative, with larger bandwidth and unprecedented peak data

rates [2]. Examples of mmWave technologies include the V-band (60 GHz) and E-band (70/80 GHz), and backhaul links using these bands may be well suited to supporting 5G due to their 10 to 25 Gbps throughput and low latency.

This paper provides a solution that models the configuration and performance for a 60 GHz mmWave 5G backhaul mesh network based on IEEE 802.11ad. The network is deployed as a service in the Liverpool 5G project to support the development of novel eHealth use cases and applications. More specifically, the paper discusses the viability of IEEE 802.11ad point-to-point links as a backhaul network in urban deployments. Our experiments largely validate the expected link performance based on the simulation parameters of distance, coding scheme and link quality but show that significant variability is introduced as a result of the real-world deployment. As such, some links match or even exceed the simulated performance, while others under-perform. These results will provide important feedback for other 5G deployments based on 60 GHz technology.

The remainder of the paper is structured as follows. First, a discussion related to work on private 5G deployments is provided in Section 2, and this is followed by a detailed description of the 5G topology deployed in the Liverpool 5G network. In Section 3, technical background related to the particular problems studied in this work are discussed, which includes a brief description of mmWave systems, modulation and coding schemes and link adaption, along with the general metrics to evaluate link performance. Following this, the process on how the scenarios have been modelled and implemented in MATLAB, including the simulation assumptions, link parameters and traffic model are discussed in Section 4. In Section 5, the simulation results showing the performance and efficiency of the 5G backhaul network are presented and discussed. Finally, Section 6 concludes this work.

## 2. Private and Community 5G Networks

### 2.1. 5G Non-Public Networks and Urban Deployments

The use of 5G technologies to support deployments beyond traditional MNO networks opens a wide range of applications and scenarios for wireless technologies. One area that has seen massive interest to date is in support of IoT-based deployments and particularly around Industry 4.0. The next generation of industrial services, from manufacturing to logistics, are being designed to heavily incorporate data networks to support a wide variety of sensor and communication requirements, where traditional wired technologies are either too expensive or infeasible to deploy on this scale. For example, even in a moderately sized factory site of 1–2 km<sup>2</sup>, the sheer scale of connectivity requirements would introduce significant problems, notwithstanding the need for flexibility in the case of reconfigurations and the potential to deploy this in a challenging or hazardous environment. As such, the use of 5G wireless technologies in this space provides many advantages including reduced cost and complexity while still providing good bandwidth and responsiveness. In the 3GPP scope, these are called private networks or non-public networks (NPNs) [3] as opposed to the traditional Public Land Mobile Networks (PLMNs).

These NPN 5G deployments could make use of a combination of shared and private base stations and backend infrastructure through slicing to provide network capacity solely provisioned for the use case. This area is still under active research in academia and beyond but an overview of some potential solutions are described here to provide context for our work. In order to support 5G NPNs, various scenarios have been envisioned whereby the 5G Core (5GC) can be provisioned and controlled locally or remotely, or through some combination of the above [4]. Essentially, both the Control Plane (CP), which provides device and network control such as access control and management, session management, mobility management and policy management, and the User Plane Functions (UPF), which deal with data routing and forwarding, need to be provisioned for the user in order to provide a 5G service [5]. On the one hand, an operator might choose to deploy just the gNB locally (or virtually through slicing) to save CAPEX and OPEX overheads and handle most of the backend functionality remotely, which greatly simplifies operation at the potential cost of some performance overheads. Conversely, a full local deployment would certainly

be more expensive and complex to operate but would be solely provisioned and therefore more performant [6].

Moreover, the heterogeneous nature of available RAN technologies in 5G provides a powerful mechanism to support a range of devices and applications, from LoRa sensor devices, through to 4K streaming or VR-type users. In addition to 5G NR, an operator might make use of some version of LTE in addition to Wi-Fi, mmWave and other technologies to provide appropriate coverage [7]. Through splitting the RAN into a Central Unit (CU) and multiple Distributed Units (DUs), different levels of control can be applied. These functional splits dictate where in the stack the separation between the CU and DUs occurs and potentially offer a great deal of flexibility in terms of how sessions are managed in the RAN, from a very 'low' traditional split up to more innovative but complex 'high' splits [8].

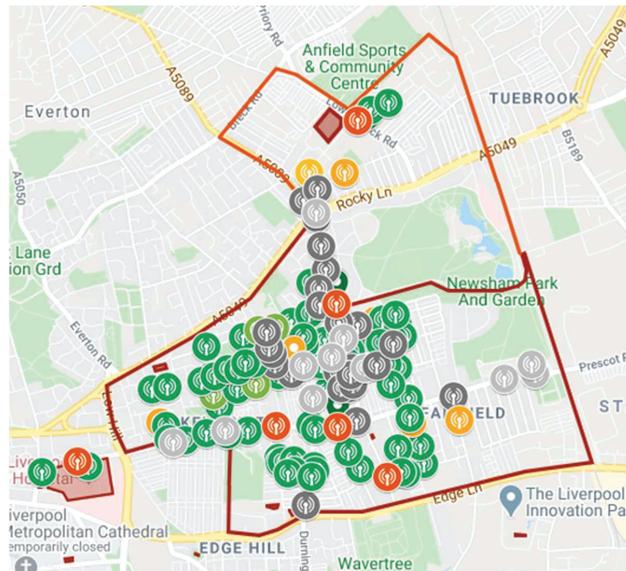
Another interesting deployment area for 5G is in public networks deployed by and for a community [7]. Such networks can, for example, be setup by a local government or organization and provided to members where MNO coverage is not available or suitable or focused around a specific use case that meets a public need. In this case, there is again a wide range of potential deployment models that could be adopted depending on the specific circumstances. An organization might apply for a specific portion of the available spectrum in their area to deploy their own base stations, or adopt another unlicensed solution. For example, a town/city with available fiber infrastructure (perhaps to support a CCTV platform) could simply look to extend that into the necessary areas by using a Wi-Fi mesh or use more specialized and higher capacity technologies as needed. It is in this context that we introduce the Liverpool 5G network, which has been deployed as part of the DCMS 5G Testbeds and Trials Programme [9].

## 2.2. Overview of the Liverpool 5G Network

The infrastructure deployed in the Liverpool 5G project is illustrated in Figure 1. In the first phase of the project (from 2018 to 2020), the network topology consisted of 34 nodes and three gateways (POPs) installed in the Kensington area of the city. Each node in the backhaul mesh was collocated with a Wi-Fi AP to provide WLAN connectivity, with some nodes also supporting ZigBee low-power ad hoc networking. The network was designed in such a way that there was a line of sight link along any of the roads between deployed nodes and the connection was based on a mmWave link (IEEE 802.11ad). The nodes are Blu-Wireless DN-101LC stations and are attached to street lights or other street furniture at a consistent elevation. The nodes have 90 degree azimuth coverage, one independent beam and a maximum capacity of 5Gbps (Mac layer). The POP nodes are connected to the gateway through a fiber link, and backhauling is then handled via a local cloud service provider. Furthermore, due to the widely deployed nodes and the high path and penetration loss at 60 GHz, some links may require relays (multi-hop transmissions) to accomplish backhauling, and nodes are clustered around the nearest POP. In the current version of the network (2020–2022), this deployment has been extended to cover a wider area and provide more ubiquitous coverage. Each node is also collocated with a 5G small cell that will run in the N77 band and standalone mode. As such, any user with a compatible handset should receive connectivity within the Kensington area of Liverpool.

The aim of this deployment is to support a range of social and healthcare applications for the community, from simple health sensor and monitoring services through to VoIP, full HD or 4 K streaming for remote consultations and (potentially) low latency VR tools. As such, the network must be able to meet a challenging set of requirements. Of course, it is very difficult to guarantee a high rate of transmission (or a small latency) in conjunction with highly reliable packet delivery (small packet error rate), due to the random transmission errors caused by the unpredictable behaviour of this type of wireless channel. Specifically, when the link qualities in the network are poor, packets may be retransmitted several times across hops in order to reach their destination. This could result in aggregation and queuing of packets at the core relay nodes, which translates to unreasonably large average end-to-end delay, as well as a low rate of transmission. However, in order to determine

the optimal operating point and achievable Quality of Service offered by the network, it is important to analyse the trade-off between the throughput, latency and reliability (PER and PDF) to improve the overall network performance [10]. Finally, average latency, which is defined as the average time from when the packet transmission starts at the source station to when the packet is correctly received by the destination [11], is a particular issue for applications which have strict real-time requirements such as video conferencing or VoIP calls. The next section takes a more detailed look at the characteristics of communications at 60 GHz to provide a foundation for our analysis work.



**Figure 1.** Overview of Liverpool 5G Backhaul Deployment.

### 3. Communication in the 60 GHz Band

The use of the 60 GHz frequency for wireless communication goes back to 2001 when the US regulator (FCC) adopted rules for unlicensed operations in the 57 to 64 GHz band for commercial and public use. Radios operating in the 60 GHz band have some unique characteristics that make them different to radios operating in the traditional 2.4/5 GHz bands. Using 60 GHz leads to smaller sizes of RF components, enabling a more compact realization of an array structure, which in turn offers larger antenna gain with high directivity. Furthermore, oxygen absorption attenuates 60 GHz signals such that they cannot travel far beyond their intended path. These properties can also help reduce interference among terminals, enhance data security and, very importantly, enable the spectrum to be re-used in dense deployment scenarios. The 60 GHz band also allows very high data rate communication in applications such as video conferencing, media streaming and gaming.

Zhen Gao et al. in [12] stated that mmWave is suitable for backhaul links in ultra-dense wireless networks due to several unique properties: having a high capacity, being inexpensive and small form factor equipment, and having an immunity to interference. More recently, research has been conducted on the use of this technology to support densely meshed wireless backhuls as shown in Figure 2. The work in [13] studied the performance of a mmWave meshed backhaul deployment in a district of Barcelona. The links between nodes in the network were based on IEEE 802.11ad technology at 60 GHz. The work demonstrated the viability of the technology over multiple hops and showed the

influence the number of gateways (POPs) and the number of radio channels available has on performance.

Further work in [14] addresses the challenges and the properties of mmWave communications for 5G to support the re-design of protocols and architectures; specifically, interference management and spatial re-use. For a high-level view of 5G fronthaul and backhaul wireless transport over mmWave the reader is referred to [15]. To further promote the development of mmWave communications, many other projects are currently evaluating this technology in the UK including the Liverpool 5G Testbed [16], the 5G Smart Tourism project [17] and the Worcestershire 5G Consortium Overview project [18], to name only three.

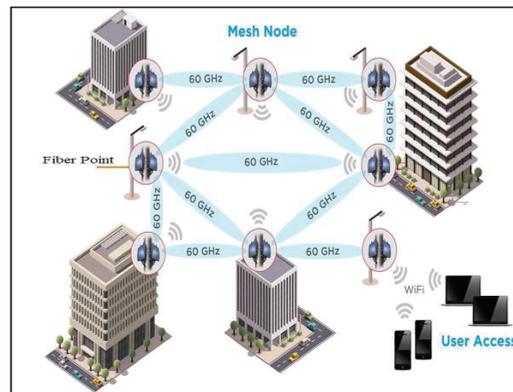


Figure 2. Example of wireless backhaul network.

IEEE 802.11ad, also known as WiGig, is an enhancement to the 802.11 standard that enables multi-gigabit wireless communication in the 60 GHz unlicensed band [19]. The current IEEE 802.11ac standard can support a maximum of 2.5 Gbps with three 160 MHz channels and 256-QAM data rate. In contrast, the 60 GHz band could provide a maximum throughput of up to 7 Gbps. However, utilizing this spectrum resource comes with challenging propagation characteristics such as strong attenuation from obstacles and huge path loss. Furthermore, due to severe penetration loss and reflection due to the short wavelength, mmWave communications are generally only feasible in line of sight (LoS) environments. To overcome these drawbacks, IEEE 802.11ad provides a robust modulation and coding scheme and link adaptation mechanism to optimize throughput and minimize packet and bit error rate. In addition, the standard defines a directional communication scheme that takes advantage of beamforming antenna gain to cope with the increased attenuation in the 60 GHz band [20].

### 3.1. Modulation and Coding Scheme

The modulation and coding scheme (MCS) index value summarizes the modulation type (e.g., BPSK, QPSK, 16QAM) and the coding rate that is used in a given physical resource block (PRB). Typically, a higher MCS index offers a higher spectral efficiency (which translates to a higher potential data rate) but requires a higher SNR to support it. Depending on the link quality metrics (LQMs) a node will dynamically choose an appropriate MCS in order to provide the best possible performance.

In the 802.11ad specification, three different PHY modes are defined based on how they can be used. The Control PHY (MCS0) is designed for low SNR operation with low throughput communication (27.5 Mbps) and is mainly used during the beamforming training (BF) phase. The Single Carrier (SC) PHY enables a power efficient and low complexity transceiver implementation. It provides a good trade-off between average throughput and energy efficiency. SC-PHY defines MCS 1–12, of which MCS 1–4 are mandatory modes to

be implemented in all devices for interoperability [21]. MCS 13–24 provides the maximum 802.11ad data rates up to 6.76 Gbps and adopts orthogonal frequency-division multiplexing (OFDM) technology, which is very efficient in multipath environments [21]. However, its implementation is complex and therefore targets devices with less stringent power and design constraints. Finally, the DMG low-power SC-PHY with MCS 25–31 is an optional SC mode that can provide lower processing power by using Reed–Solomon instead of low-density parity check (LDPC) codes [22,23].

In this work we consider the SC-PHY model, which ranges in value from MCS 1 to MCS 12. Table 1 below lists the MCS values defined in the IEEE 802.11ad standard and gives their corresponding modulation schemes, coding and data rates. In the SC-PHY model, the lowest SC data rate is 385 Mb/s (MCS 1), which is implemented using BPSK modulation and rate 1/2 code with a symbol repetition of two. MCS 1–5 are all based on  $\pi/2$ -BPSK modulation. MCS 2, 3, 4 and 5 use code rate 1/2, 5/8, 3/4 and 13/16, respectively. MCS 6–9 are based on  $\pi/2$ -QPSK modulation, whereas MCS 10–12 are based on  $\pi/2$ -QPSK [23].

**Table 1.** SC\_PHY Modulation and Coding Schemes.

MCS Index	Modulation	Code Rate	PHY Rate (Mbps)	Notes
1	$\pi/2$ -BPSK	1/2	385	Single Carrier Phy 2 repeated frames
2	$\pi/2$ -BPSK	1/2	770	Single Carrier Phy
3	$\pi/2$ -BPSK	5/8	962.5	Single Carrier Phy
4	$\pi/2$ -BPSK	3/4	1155	Single Carrier Phy
5	$\pi/2$ -BPSK	13/16	1251.2	Single Carrier Phy
6	$\pi/2$ -QPSK	1/2	1540	Single Carrier Phy
7	$\pi/2$ -QPSK	5/8	1925	Single Carrier Phy
8	$\pi/2$ -QPSK	3/4	2310	Single Carrier Phy
9	$\pi/2$ -QPSK	13/16	2502.5	Single Carrier Phy
10	$\pi/2$ -16-QAM	1/2	3080	Single Carrier Phy
11	$\pi/2$ -16-QAM	5/8	3850	Single Carrier Phy
12	$\pi/2$ -16-QAM	3/4	4620	Single Carrier Phy

### 3.2. Link Adaptation

In wireless communication systems, the quality of a wireless signal received by nodes depends on a number of factors: the distance between the nodes, path loss exponent, log-normal shadowing, short term (Rayleigh) fading and noise. In order to improve system capacity and peak data rate, the signal transmitted to and by a particular node is modified to account for the signal quality variation through a process commonly referred to as link adaptation. This is also known as adaptive modulation and coding (AMC) techniques [24].

In communication systems, the use of AMC techniques allows the system to achieve higher spectral efficiency by dynamically changing the modulation and coding schemes based on the channel statistics so as to improve overall spectral efficiency. In other words, it is utilized to set the modulation and coding in order to reflect the features of the wireless link and to maximize throughput. Moreover, AMC has been widely used to match transmission parameters to time-varying channel conditions in order to enhance the spectral efficiency while adhering to a target error performance over wireless channels [25,26].

A number of research works have been conducted on link adaptation, and new link adaptation schemes have been proposed. Holland et al. [27] introduced a Receiver-Based Auto-Rate (RBAR) protocol based on the Request-To-Send (RTS) and Clear-To-Send (CTS) mechanism by adjusting the IEEE 802.11 standard. The basic idea of RBAR is that the receiver estimates the wireless channel quality using a sample of the instantaneously received signal strength at the end of the RTS reception, then selects the appropriate

transmission rate based on this estimate, which feeds back to the transmitter, and finally the transmitter responds to the receipt of the CTS by transmitting the data packet at the rate chosen by the receiver [28]. Kamerman et al. [29] presented the Auto Rate Fall-back (ARF) protocol for IEEE 802.11, which is used in Lucent's WaveLAN devices. With ARF, the sender selects the best rate based on information retrieved from previous data packet transmissions and incrementally increases or decreases the rate after a number of consecutive successes or losses.

### 3.3. Beamforming

Beamforming refers to a technique that dynamically shapes the beam pattern to focus on specific directions. It is a spatial filtering technique used in smart antennas and the main objective is to maximize the received power directed towards a certain node while minimizing the interference power towards undesired nodes [30]. A signal processor controls the excitation of antenna array elements to synthesize a desired radiation pattern [31]. In other words, beamforming works by combining elements in a phased array in such a way that signals at particular angles experience constructive interference (at the main lobe), while others experience destructive interference (at the nulls) and at the receiver by having gains in one direction and attenuation in others.

The beamforming technique is used in smart antennas for transmitting and receiving signals in massive multiple-input multiple-output (MIMO) systems. MIMO systems combined with beamforming antenna array technologies are expected to play a key role in 5G wireless communication systems [30]. Apart from a higher directive gain, these antennas offer complex beamforming capabilities that increase the capacity of networks by improving the signal-to-interference ratio (SIR).

Beamforming is mandatory in 802.11ad, and both transmitter-side and receiver-side beamforming is supported. In the 802.11ad standard, beamforming training determines the appropriate receive and transmit antenna sectors for a pair of nodes. The beamforming is split into two phases. The sector level sweep (SLS) and beam refinement phase (BRP). During SLS, an initial coarse-grained antenna sector configuration is determined. Thus, in the SLS phase each of the two nodes either trains or receives the appropriate transmit antenna sector. This information is used in a subsequent optional BRP to fine-tune the selected sectors [20]. During this stage, antenna weight vectors that vary from predefined sector patterns are evaluated to further optimize transmissions on phased antenna arrays.

### 3.4. Performance Evaluation

As part of the deployment and operation of the 5G network, an evaluation of the mmWave link performance was undertaken to better understand the characteristics of 802.11ad when it is deployed in this manner. The outcomes of this work will then be used to optimize the clustering and link level configuration of the network going forward. Ultimately, our goal is to develop dynamic optimization algorithms that could be used as part of a self-optimizing network (SON) managed by Software-Defined Networking (SDWN). This algorithm could use information on link characteristics from across the meshed backhaul, along with current monitoring statistics on traffic load, to reconfigure links and optimize routing to perform load balancing and help enforce user QoS.

To date we have undertaken preliminary work to evaluate the optimal clustering and channel selection based on simple Dijkstra shortest path routing algorithms and calculated the optimal link distance based on the expected 802.11ad performance using packet error rate in a typical 'street canyon' deployment such as an urban environment. This means that we are now in a position to predict link performance for a given deployment and alignment of nodes. Our aim for the work presented in this paper was, therefore, to model a real node deployment as part of the latest phase of the project, which began in September 2021, and then verify our predicted performance using monitoring information once it was operational. The modelled deployment will be explained in detail in the next section.

#### 4. Simulation Model

We consider a case study to simulate the deployment of the backhaul network on Sheil Road in Liverpool, by the Fairfield Medical Centre. This deployment was developed based on a 5G network planning tool, which provides an online copy of Kensington in Liverpool, for the Liverpool 5G network (<https://www.cgasimulation.com/network-planning-tool/>, last accessed: 15 December 2021). From this tool, we were able to build a simulated deployment of nodes and model their performance in an urban environment [32]. Figure 3 illustrates the Fairfield Medical Centre modelled by the planning tool. Moreover, the tool provides information for each of the nodes in the network, such as the site of the node, the kind of installation, the direction, the latitude and the longitude, which we used to build our simulation.



Figure 3. Copy of the Fairfield Medical Centre deployment.

Using these sites, we were able to obtain details on the link distances and orientations to build a MATLAB-based simulation of this end-to-end deployment that allowed us to carry out the performance evaluation illustrated in the next section. These sites are simulated based on IEEE 802.11ad, which defines a directional multi-gigabit (DMG) transmission format operating in the unlicensed band around 60 GHz, using single carrier (SC) PHY link [32]. The channels, on the other hand, are modelled as a multipath fading channel using the model environment of ‘street canyon hotspot’ [33].

We evaluated the individual link performances in terms of their Packet Error Rate (PER) as a function of the signal-to-noise ratio and for different modulation and coding scheme values. Specifically, first, a set of SNR points in dB were selected based on each simulated MCS. Second, for each SNR point, multiple packets were transmitted through a TGay millimeter wave channel [34], then synchronized and demodulated, and the received Physical Layer Convergence Procedure Service Data Units (PSDUs) were recovered. Third, the received PSDUs were compared to those transmitted to determine the number of packet errors and hence the PER. The number of packets considered to compute the PER for each SNR point depends on the following parameters: (1) the maximum number of packet errors simulated at each SNR point. When the packet error number reaches this limit, the simulation at this SNR point is finalized; (2) the maximum number of packets simulated at each SNR point, which limits the length of the simulation if the packet error limit is not reached. In order to obtain meaningful results, we considered 100 and 1000 as the maximum number of packet errors and maximum number of packets, respectively [34].

## 5. Simulation Results

### 5.1. Evaluation in MATLAB

To evaluate the link performance of the Sheil Road deployment, we first simulate the network layout using MATLAB as shown in Figure 4, based on the information identified in the previous section. The MATLAB simulator is built to accurately model the properties and parameters of IEEE 802.11ad networks and closely represents the deployment in Liverpool 5G Create in terms of node position and alignment. Specifically, in Figure 4 there are three links used to connect three IEEE 802.11ad access points (APs), i.e., nodes 2–4, with node 1 acting as a gateway according to the 5G Create deployment illustrated in Figure 3. Moreover, links that are configured to work on the same channel have the same colour in the figure, i.e., red for Channel 2 at 60.48 GHz and blue for Channel 3 at 62.64 GHz. Table 2 summarizes the main simulation parameters [32–34].

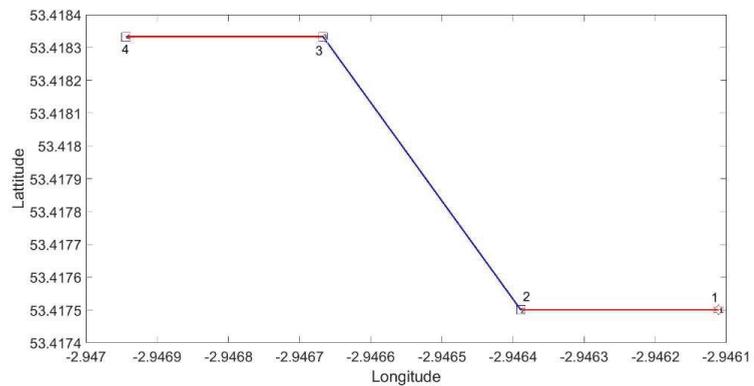


Figure 4. Network Layout.

Table 2. Simulation Parameters [32–34].

Parameter	Value
Network Standard	IEEE 802.11ad
Radio Channels	60.48 GHz, 62.64 GHz
Pathloss Model	Street Canyon Hotspot
MCSs	7–12
Maximum number of packet errors for computation of PER	100
Maximum number of packet for computation of PER	1000

We next present the performance of the longest link, i.e., Link 2 from node 2 to node 3 in Figure 4, which is 94.4 m long, and one of the shortest links, i.e., Link 3 from node 3 to node 4, which is 18.35 m long. As introduced in the previous section, the PER of these links was computed as a function of the SNR and for different MCS values as shown in Figure 5 for Link 2 and Figure 6 for Link 3, respectively. As can be noted in both figures, the PER represented in log scale decreases as the SNR value increases for all MCS types. In addition, higher MCS can be achieved for a targeted level of PER when the SNR of the considered link increases. For instance, in the case of very low targeted levels of PER, in Link 2 MCS 11 can be achieved when the PER in percentage is 1.3% and the SNR is 20dB, while MCSs 8 and 9 can be considered in the case of targeting a PER of 1% and 0.2%, and SNR of 14 dB and 15 dB, respectively. In Link 3, in the case of low targeted levels of PER, MCSs 11, 8 and 9 can be achieved when the PERs are 2.5%, 0.8% and 0.57%, and the SNRs are 21dB, 16 dB and 18.5 dB, respectively.

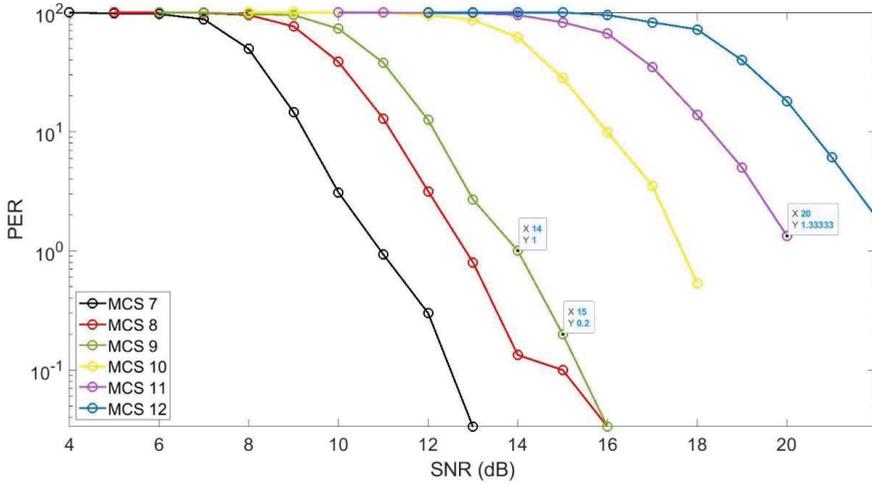


Figure 5. Link2 performance.

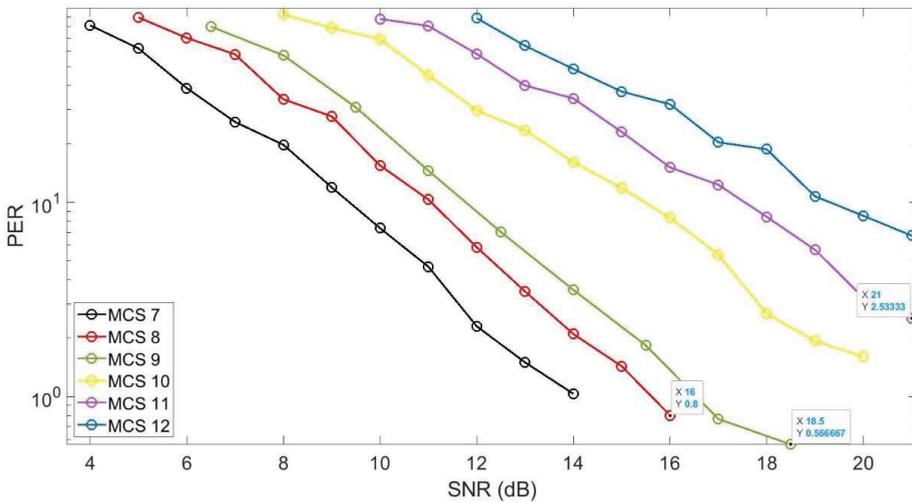


Figure 6. Link3 performance.

5.2. Validation through GRAFANA

In order to validate the link performance of the Sheil Road deployment illustrated above, we monitored Link 2 and Link 3 through a Grafana-based interface for 7 days. The testbed allows us to monitor and visualize in real time the link performance of the backhaul network at Sheil Road illustrated in Figure 3. Note that each node of the deployment is characterized by an ID defined in Grafana that will be illustrated together with the monitored results.

Specifically, Link 2 connects the node with ID sheil-rd-003-f8627 to the node with ID sheil-rd-004-f3080, represented as Node 2 and Node 3 in Figure 4, respectively. Figures 7–9 illustrate, among other parameters, the SNR (green line in Figure 7), the performance in terms of PER (red line in Figure 8) and the MCSs (yellow circles in Figure 9) at the receiver node, i.e., node sheil-rd-004-f3080. Other parameters that can be monitored through Grafana and that we do not illustrate because they are out of the scope of this paper are:

the Received Channel Power Indicator (RCPI) (blue line in Figure 7); the Automatic Gain Control (AGC) (orange line in Figure 7); the MAC transmit rate (Tx Rate) (green line in Figure 8); and the Modulation error ratio (MAR) (lilac line in Figure 8). From these figures we can note that some parameters present a high oscillation amplitude and, therefore, we also included the average value of each parameter at the bottom of the diagrams. In summary, we can conclude that the average SNR is 14 dB, the PER on average is 0.18% and the average MCS value is 9.

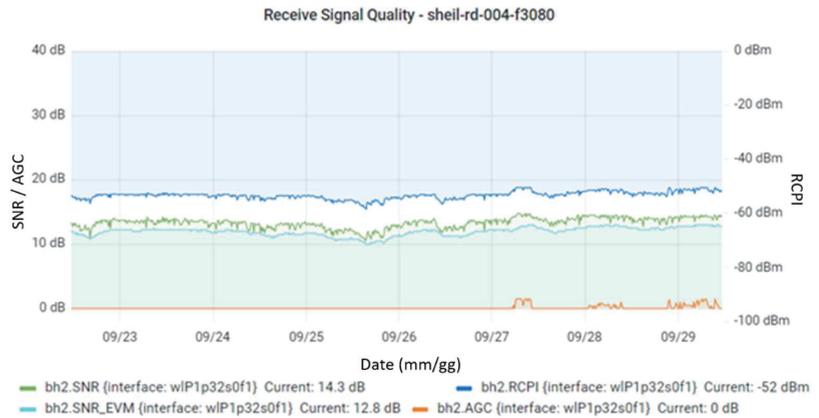


Figure 7. Performance in terms of SNR of Link 2.

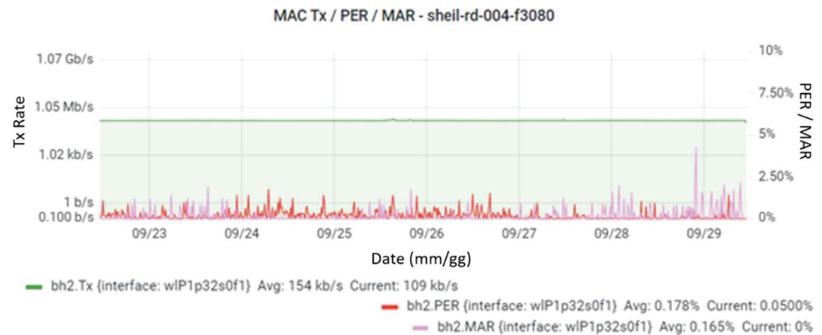


Figure 8. Performance in terms of PER of Link 2.

Link 3 connects the node with ID sheil-rd-005-f8627 to the node with ID sheil-rd-006-spfhc represented as Node 3 and Node 4 in Figure 4, respectively. Figures 10–12 illustrate again, among other parameters, the SNR (green line in Figure 10), the performance in terms of PER (red line in Figure 11) and the MCSs (yellow circles in Figure 12) at the receiver node, i.e., node sheil-rd-006-spfhc. From these figures we can conclude that the average SNR is 18.5 dB, the PER on average is 0.56% and the average MCS is 11.

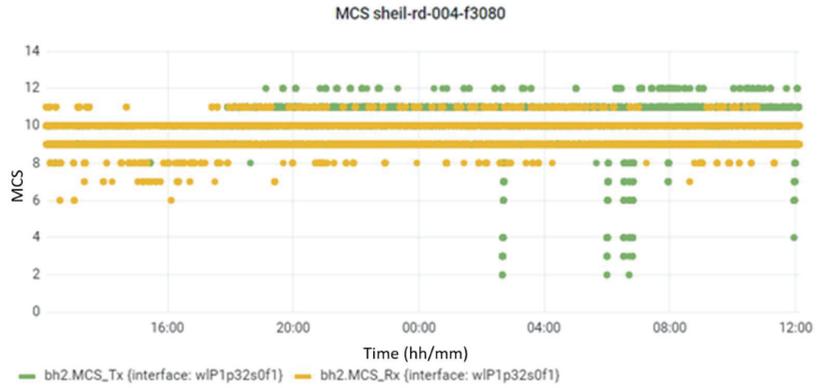


Figure 9. Performance in terms of MCS of Link 2.

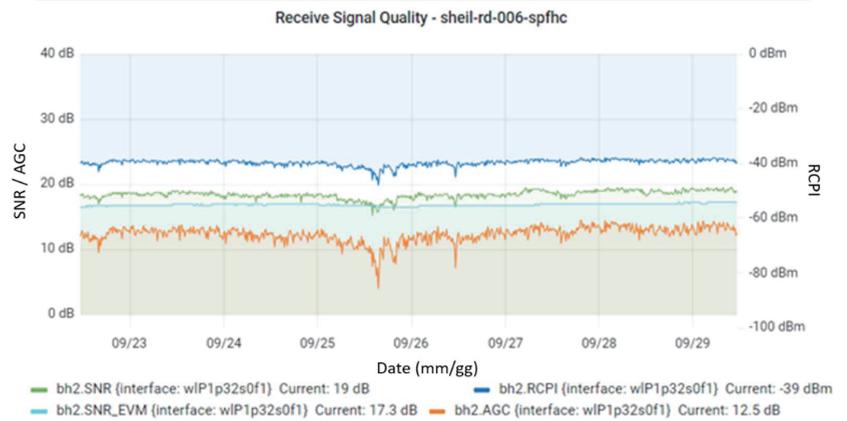


Figure 10. Performance in terms of SNR of Link 3.

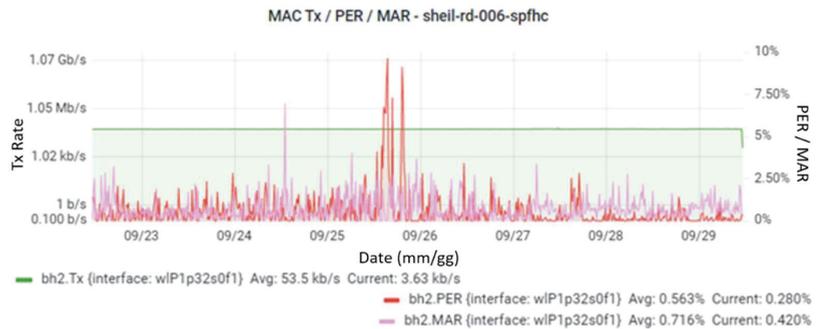


Figure 11. Performance in terms of PER of Link 3.

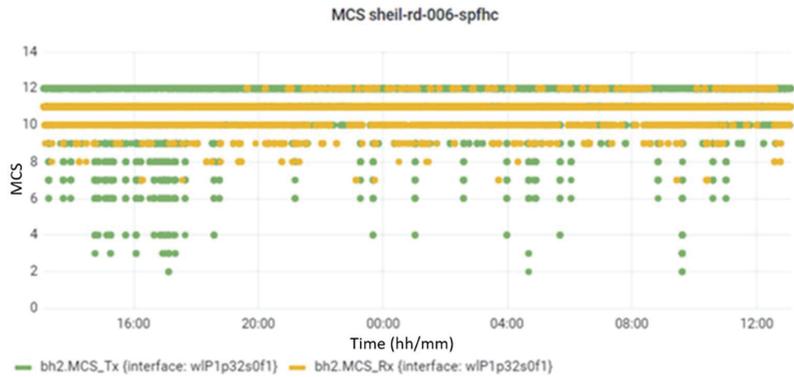


Figure 12. Performance in terms of MCS of Link 3.

5.3. Discussion

Table 3 summarizes all the results illustrated in the previous subsections. Specifically, based on the monitoring carried out through Grafana, in the table we can observe that the longest link achieves an SNR of 14 dB and can use MCS 9 guaranteeing a PER of approximately 0.2%. On the other hand, based on the simulations, the link is characterized by the same PER and MCS when it can achieve 15 dB SNR. Note that radio-to-radio variations of 1–2dB on the SNRs monitored through Grafana are expected due to manufacturing tolerances and, therefore, the longest link modelled through MATLAB can be considered validated. Moreover, in the table we can observe that the short link, based on the monitored information, achieves an SNR of 18.5 dB and can use MCS 11 guaranteeing a PER of approximately 0.6%. On the other hand, based on the simulations, this link in the case of 18.5 dB SNR can use MCS 9 in order to guarantee the same value of PER. Note that, again considering a possible radio-to-radio variation of 1–2 dB on the monitored values, the short link could reach MCS 10 guaranteeing the same value of PER computed through Grafana. Therefore, even taking into account monitoring errors due to manufacturing tolerances, there is still a reduction in terms of maximum available throughput of 20% in the short link, which can provide 3080 Mbps in the simulator and 3850 Mbps in Grafana (see Table 1).

Table 3. Results Comparison.

	MATLAB			Grafana		
	SNR (dB)	PER (%)	MCS	SNR (dB)	PER (%)	MCS
Long Link 2–3	20	1.3	11	14	0.18	9
	15	0.2	9			
	14	1	8			
Short Link 3–4	21	2.5	11	18.5	0.56	11
	18.5	0.57	9			
	16	0.8	8			

In summary, our simulation results, therefore, provide a variable picture in terms of corroborating the performance for the short link. One explanation of this could be that the actual nodes are performing better under certain circumstances and this needs to be programmed into our simulator. Other explanations include minor differences in the physical deployment of the equipment (e.g., alignment) or environmental factors that affect link performance. These results, therefore, generally validate our simulations but highlight the issues in predicting link performance in such a dynamic environment and

with a limited set of links to evaluate. Our immediate next steps are, therefore, to gather more data on the configuration and performance of the nodes as the deployment continues and monitor them further to refine our models over time.

## 6. Conclusions and Further Work

The popularity of wireless communications for a wide range of use cases means that 5G technologies are likely to become increasingly ubiquitous over time, both in traditional MNO deployments but also in a range of other scenarios, from 5G NPN for supporting IIoT to community-owned networks. However, the vast range of technologies and techniques that represent 5G, both in the core and at the edge, make it very complicated to define exactly what 5G ‘looks like’ and how it performs in every situation.

In Liverpool, a 5G network has been deployed by a consortium including the local council and healthcare providers, technology firms and academic institutions to support healthcare and community care services. This network leverages the existing CCTV fibre network and extends it using mmWave backhaul links into areas of the city where connectivity is required. These backhauls are built using 802.11ad point-to-point links that offer up to multi-gigabit services but are susceptible to the restrictions of 60 GHz communications.

In an effort to understand this better, we built a simulator in MATLAB to model the behaviour of these links under a range of conditions, and compared our results to the real-world deployments to validate our findings. Our results show that some links conform to our model very well, while others are somewhat unpredictable. This difference could be due to manufacturing tolerances or variations in link quality over short distances. This is a valuable contribution to ongoing 5G deployment efforts that utilize this technology, as real-world characteristics for 60 GHz link performance will help to refine node placement and configuration. It also provides useful insights to the research community on the effectiveness of novel private and community 5G-utilizing heterogeneous technologies.

Over time, we aim to continually improve our models through further validation to generalize our model such that it can be used as a general-purpose tool for network planning or as a basis for further work to examine network optimization. We would also be interested in evaluating the potential impact of 802.11ay-based links in this environment and conducting a comparative analysis of their performance.

**Author Contributions:** Conceptualization, M.M. and A.R.; methodology, M.M., A.R. and O.T.; software, O.T. and A.R.; validation, O.T. and A.R.; formal analysis, O.T. and A.R.; investigation, M.M. and A.R.; writing—original draft preparation, M.M. and A.R.; writing—review and editing, M.M. and A.R.; visualization, O.T.; supervision, M.M. and A.R.; project administration, M.M.; funding acquisition, M.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable, the study does not report any data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Kodjo, A. *Design and Optimization of Wireless Backhaul Networks*; Université Nice Sophia Antipolis: Nice, France, 2014.
2. Kim, J.; Kim, J.-K. Achievable rate estimation of IEEE 802.11 ad visual big-data uplink access in cloud-enabled surveillance applications. *PLoS ONE* **2016**, *11*, e0167447. [[CrossRef](#)]
3. TTrakadas, P.; Sarakis, L.; Giannopoulos, A.; Spantideas, S.; Capsalis, N.; Gkonis, P.; Karkazis, P.; Rigazzi, G.; Antonopoulos, A.; Cambeiro, M.A.; et al. A Cost-Efficient 5G Non-Public Network Architectural Approach: Key Concepts and Enablers, Building Blocks and Potential Use Cases. *Sensors* **2021**, *21*, 5578. [[CrossRef](#)]
4. Rostami, A. Private 5G Networks for Vertical Industries: Deployment and Operation Models. In Proceedings of the IEEE 5G World Forum (5GWF), Dresden, Germany, 30 September–2 October 2019; pp. 433–439. [[CrossRef](#)]
5. Aijaz, A. Private 5G: The Future of Industrial Wireless. *IEEE Ind. Electron. Mag.* **2020**, *14*, 136–145. [[CrossRef](#)]

6. Strinati, E.C.; Haustein, T.; Maman, M.; Keusgen, W.; Wittig, S.; Schmieder, M.; Barbarossa, S.; Merluzzi, M.; Klessig, H.; Giust, F.; et al. Beyond 5G Private Networks: The 5G CONNI Perspective. In Proceedings of the 2020 IEEE Globecom Workshops (GC Wkshps), Taipei, Taiwan, 7–11 December 2020; pp. 1–6. [\[CrossRef\]](#)
7. Patwary, M.N.; Nawaz, S.J.; Rahman, A.; Sharma, S.K.; Rashid, M.; Barnes, S.J. The Potential Short- and Long-Term Disruptions and Transformative Impacts of 5G and Beyond Wireless Networks: Lessons Learnt From the Development of a 5G Testbed Environment. *IEEE Access* **2020**, *8*, 11352–11379. [\[CrossRef\]](#)
8. Larsen, L.M.P.; Checko, A.; Christiansen, H.L. A Survey of the Functional Splits Proposed for 5G Mobile Crosshaul Networks. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 146–172. [\[CrossRef\]](#)
9. Liverpool 5G Health and Social Care Testbed, Overview, January 2020. Available online: <https://liverpool5g.org.uk/wp-content/uploads/2021/07/Liverpool-5G-Health-and-Social-Care-Testbed-Overview.pdf> (accessed on 15 December 2021).
10. Srinivasa, S.; Haenggi, M. Throughput-delay-reliability tradeoffs in multihop networks with random access. In Proceedings of the 2010 48th Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, 29 September–1 October 2010; pp. 1117–1124. [\[CrossRef\]](#)
11. Brienza, S.; de Guglielmo, D.; Anastasi, G.; Conti, M.; Neri, V. Strategies for optimal MAC parameter setting in IEEE 802.15.4 wireless sensor networks: A performance comparison. In Proceedings of the 2013 IEEE Symposium on Computers and Communications (ISCC), Split, Croatia, 7–10 July 2013; pp. 898–903.
12. Gao, Z.; Dai, L.; Mi, D.; Wang, Z.; Imran, M.A.; Shakir, M.Z. MmWave massive-MIMO-based wireless backhaul for the 5G ultra-dense network. *IEEE Wirel. Commun.* **2015**, *22*, 13–21. [\[CrossRef\]](#)
13. Legg, P.; McConnell, R. Meshed Backhauling of Small Cells Using IEEE802.11ad at 60GHz. In Proceedings of the 2018 European Conference on Networks and Communications (EuCNC), Ljubljana, Slovenia, 18–21 June 2018; pp. 393–397.
14. Seker, C.; Güneser, M.T.; Ozturk, T. A Review of Millimeter Wave Communication for 5G. In Proceedings of the 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Ankara, Turkey, 19–21 October 2018; pp. 1–5.
15. Kuo, P.-H.; Mourad, A. Millimeter wave for 5G mobile fronthaul and backhaul. In Proceedings of the 2017 European Conference on Networks and Communications (EuCNC), Oulu, Finland, 12–15 June 2017; pp. 1–5.
16. Liverpool 5G Health and Social Care Testbed, Developing the Network, January 2020. Available online: <https://liverpool5g.org.uk/wp-content/uploads/2021/07/Liverpool-5G-Health-and-Social-Care-testbed-Developing-the-Network.pdf> (accessed on 15 December 2021).
17. UK5G. The 5G Smart Tourism Project. 2019. Available online: <https://uk5g.org/discover/testbeds-and-trials/5g-smart-tourism/> (accessed on 15 December 2021).
18. W. 5G. Worcestershire 5G Testbed Project. 2019. Available online: <https://www.wlep.co.uk/current-projects/worcestershire-5g/> (accessed on 15 December 2021).
19. Sakaguchi, K.; Mohamed, E.M.; Kusano, H.; Mizukami, M.; Miyamoto, S.; Rezagah, R.E.; Takinami, K.; Takahashi, K.; Shirakata, N.; Peng, H.; et al. Millimeter-Wave Wireless LAN and Its Extension toward 5G Heterogeneous Networks. *IEICE Trans. Commun.* **2015**, *98*, 1932–1948. [\[CrossRef\]](#)
20. Nitsche, T.; Cordeiro, C.; Flores, A.B.; Knightly, E.W.; Perahia, E.; Widmer, J.C. IEEE 802.11 ad: Directional 60 GHz communication for multi-Gigabit-per-second Wi-Fi. *IEEE Commun. Mag.* **2014**, *52*, 132–141.
21. Schultz, B. *802.11 ad-WLAN at 60 GHz—A Technology Introduction*; Rohde & Schwarz: Munich, Germany, 2013.
22. Assasa, H.; Widmer, J. Implementation and Evaluation of a WLAN IEEE 802.11 ad Model in ns-3. In Proceedings of the Workshop on ns-3; ACM: New York, NY, USA, 2016; pp. 57–64.
23. Group, I.W. *IEEE Standard for Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments*; IEEE Std.: New York, NY, USA, 2010; Volume 802.
24. Simoens, S.; Pellati, P.; Gosteau, J.; Gosse, K.; Ware, C. The evolution of 5GHz WLAN toward higher throughputs. *IEEE Wirel. Commun.* **2003**, *10*, 6–13. [\[CrossRef\]](#)
25. Liu, Q.; Zhou, S.; Giannakis, G. Queuing with adaptive modulation and coding over wireless links: Cross-Layer analysis and design. *IEEE Trans. Wirel. Commun.* **2005**, *4*, 1142–1153. [\[CrossRef\]](#)
26. Alouini, M.-S.; Goldsmith, A.J. Adaptive Modulation over Nakagami Fading Channels. *Wirel. Pers. Commun.* **2000**, *13*, 119–143. [\[CrossRef\]](#)
27. Holland, G.; Vaidya, N.; Bahl, P. A rate-adaptive MAC protocol for multi-hop wireless networks. In Proceedings of the 7th Annual International Conference on Mobile Computing and Networking; ACM: New York, NY, USA, 2001; pp. 236–251.
28. Qiao, D.; Choi, S.; Shin, K.G. Goodput analysis and link adaptation for IEEE 802.11 a wireless LANs. *IEEE Trans. Mob. Comput.* **2002**, *99*, 278–292. [\[CrossRef\]](#)
29. Kameran, A.; Monteban, L. WaveLAN@II: A high-performance wireless LAN for the unlicensed band. *Bell Labs Tech. J.* **1997**, *2*, 118–133. [\[CrossRef\]](#)
30. Ali, E.; Ismail, M.; Nordin, R.; Abdulah, N.F. Beamforming techniques for massive MIMO systems in 5G: Overview, classification, and trends for future research. *Front. Inf. Technol. Electron. Eng.* **2017**, *18*, 753–772. [\[CrossRef\]](#)
31. Murray, B.P.; Zaghoul, A.I. A survey of cognitive beamforming techniques. In Proceedings of the 2014 United States National Committee of URSI National Radio Science Meeting (USNC-URSI NRS), Boulder, CO, USA, 8–11 January 2014; p. 1.

32. IEEE Std 802.11ad™-2012 IEEE Standard for Information Technology—Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Amendment 3: Enhancements for Very High Throughput in the 60 GHz Band; IEEE Std.: New York, NY, USA, 2013.
33. Maltsev, A.; Erceg, V.; Perahia, E.; Hansen, C.; Maslennikov, R.; Lomayev, A.; Sevastyanov, A.; Khoryaev, A.; Morozov, G.; Jacob, M.; et al. *Channel Models for 60 GHz WLAN Systems*; IEEE Standard 802.11-09/0334r8; IEEE Std.: New York, NY, USA, 2010.
34. Maltsev, A.; Pudeyev, A.; Gagiev, Y.; Lomayev, A.; Bolotin, I.; Johnsson, K.; Sakamoto, T.; Motozuka, H.; Gentile, C.; Papazian, P.; et al. *Channel Models for IEEE 802.11ay*; IEEE Standard 802.11-15/1150r9; IEEE Std.: New York, NY, USA, 2017.



Article

# On Predicting Ticket Reopening for Improving Customer Service in 5G Fiber Optic Networks

Lorenzo Ricciardi Celsi <sup>1,2,\*</sup>, Andrea Caliciotti <sup>2,3</sup>, Matteo D’Onorio <sup>4</sup>, Eugenio Scocchi <sup>5</sup>, Nour Alhuda Sulieman <sup>6</sup> and Massimo Villari <sup>6</sup>

- <sup>1</sup> ELIS Innovation Hub, Via Sandro Sandri 81, 00159 Roma, Italy
- <sup>2</sup> Department of Computer, Control, and Management Engineering Antonio Ruberti, Sapienza Università di Roma, Via Ariosto 25, 00185 Roma, Italy; andrea.caliciotti@enel.com
- <sup>3</sup> Enel Green Power S.p.A., Viale Regina Margherita 125, 00198 Roma, Italy
- <sup>4</sup> DIAEE, Sapienza Università di Roma, Corso Vittorio Emanuele II 244, 00186 Roma, Italy; matteo.donorio@uniroma1.it
- <sup>5</sup> ERG S.p.A., Via Bissolati 76, 00187 Roma, Italy; escocchi@erg.eu
- <sup>6</sup> Dipartimento di Scienze Matematiche e Informatiche, Scienze Fisiche e Scienze della Terra, Università di Messina, Piazza Pugliatti 1, 98122 Messina, Italy; nosulieaman@unime.it (N.A.S.); mvillari@unime.it (M.V.)
- \* Correspondence: l.ricciardicelsi@elis.org

**Abstract:** The paper proposes a data-driven strategy for predicting technical ticket reopening in the context of customer service for telecommunications companies providing 5G fiber optic networks. Namely, the main aim is to ensure that, between end user and service provider, the Service Level Agreement in terms of perceived Quality of Service is satisfied. The activity has been carried out within the framework of an extensive joint research initiative focused on Next Generation Networks between ELIS Innovation Hub and a major network service provider in Italy over the years 2018–2021. The authors make a detailed comparison among the performance of different approaches to classification—ranging from decision trees to Artificial Neural Networks and Support Vector Machines—and claim that a Bayesian network classifier is the most accurate at predicting whether a monitored ticket will be reopened or not. Moreover, the authors propose an approach to dimensionality reduction that proves to be successful at increasing the computational efficiency, namely by reducing the size of the relevant training dataset by two orders of magnitude with respect to the original dataset. Numerical simulations end the paper, proving that the proposed approach can be a very useful tool for service providers in order to identify the customers that are most at risk of reopening a ticket due to an unsolved technical issue.

**Keywords:** 5G fiber optic networks; data-driven service assurance; next generation networks; predictive analytics

**Citation:** Ricciardi Celsi, L.; Caliciotti, A.; D’Onorio, M.; Scocchi, E.; Sulieman, N.A.; Villari, M. On Predicting Ticket Reopening for Improving Customer Service in 5G Fiber Optic Networks. *Future Internet* **2021**, *13*, 259. <https://doi.org/10.3390/fi13100259>

Academic Editor: Michael Mackay

Received: 17 September 2021  
Accepted: 4 October 2021  
Published: 9 October 2021

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Effective customer care and data-driven service assurance have become a vital need for telecommunication companies, especially as regards the progression towards automating the management of technical tickets: in addition, this enables a thorough and objective evaluation of the performance of the service assurance functions, based on generated reports and prescribed Key Performance Indicators (KPIs), which implies increased productivity, improved quality of service and, in some cases, even personalized satisfaction of the end user.

More in detail, designing a successful data-driven service assurance system requires:

- in-house design and development in order to offer a solution that is at the same time cheap and adaptable to all service needs;
- tight integration with the customer portal in order to allow the access to all relevant online services with the same login credentials;

- support to all activities that are characteristic of an incident management process, such as opening, classifying, assigning, solving, closing and archiving an IT incident;
- enabling the exchange of files and comments regarding an IT incident, as this makes the served customers satisfied by allowing them to interact with IT agents in order to clarify the problem or understand the solution;
- provision of email notifications that allow for keeping all involved parties informed on any ticket update;
- automatic classification and ticket rerouting to the relevant IT agents, which is expected to significantly minimize resolution time of each ticket and consequently enhance user satisfaction;
- generation of reports to measure the KPIs that are chosen to evaluate the success of the customer care service.

For detailed references, the advantages and challenges of using data-driven service assurance in an organization have already been explored in the Information Technology Infrastructure Library (ITIL) framework of best practices for delivering IT services (see [1–8]). Moreover, the positive influence of ticketing services on the implementation of the incident management process is stressed in [1–10]. Indeed, being able to automate several key activities such as identification, prioritization, assignment, diagnosis and closure of technical tickets plays a relevant role in enhancing the said process. In addition, data-driven service assurance (see [4–6]) facilitates the measurement and improvement of several important KPIs for the IT business processes, such as the percentage of incidents detected and solved in the first attempt, the mean number of incidents that occurred per day and the average lifetime of an incident. In particular, in [8], it is clearly stated that automatic incident classification proves to be extremely effective at minimizing ticket resolution time. Providing an automated solution to the challenge of ticket classification is a relevant emerging task also according to [10].

Some related additional features that are in part beyond the scope of this work but testify the interest of the scientific and industrial community are the multimedia chat service architecture introduced in [11], as well as rule-based reasoning for fault diagnosis and visual dashboards for helping desk tickets monitoring, which is illustrated in [12,13]. In addition, a probabilistic framework for IT ticket annotation and search based on natural language processing is introduced in [14–17]. In [15], a predictive model based on Support Vector Machines (SVMs) and K-Nearest Neighbours (KNN) is discussed with the aim to automate incident categorization with the specific help of ticket description and other relevant ticket attributes. In a similar way, in [18], the dispatch of a ticket to the correct resolution group is successfully automated by means of a tool that combines SVMs and discriminative term-based classification techniques. Alternatively, Multinomial Naive Bayes (MNB) and Softmax Regression Neural Network (SNN) are used in [19] for text classification purposes aimed at categorizing user tickets. Finally, several methods to detect duplicate tickets/bugs are proposed in [20–22].

In particular, this paper, with respect to the emerging need for preventing customers from issuing a request for technical ticket reopening on customer service platforms of telecommunication companies, provides the following contributions:

- the identification of relevant correlations between the reopening of a technical customer service ticket, on the one hand, and Quality of Service (QoS) parameters of the 5G fiber optic networks, on the other hand, based on the actual use the customer is currently making of the fixed network itself;
- based on such correlations, the design of a data-driven model capable of predicting whether a customer will call the assistance service once again even though his/her technical ticket has already been closed.

Incidentally, reopened tickets are to be considered as those tickets that were formerly solved and have been reopened [23].

The proposed approach may prove particularly useful in the domain of 5G enabling technologies. Indeed, even with the advent of 5G, optical fiber is the most suitable means

for wireless backhaul networks. Indeed, even in networks where this is not the case, the wireless backhaul actually has to be connected into a fiber backhaul. For this reason, fiber technology is increasingly being preferred for the so-called fronthaul, especially when it comes to connecting the dense mesh of 5G small cells. There are several benefits, such as increased speeds matched with lower attenuation, significant immunity with respect to electromagnetic interference, relatively small size, and practically unlimited potential in terms of bandwidth. Hence, customer service in order to address any technical issue relative to the Quality of Service (QoS) perceived in 5G fiber optic networks has a critical role, especially with the advent of the emerging Fixed Wireless Access (FWA) paradigm [24].

The paper contribution lies in the fact that the effectiveness of the proposed approach is evaluated on the customer complaints that arise in conditions of intensive usage of the fixed network of a major Italian network operator. Indeed, according to IBM analyses in [25] and to [26], by automating up to 85% of the customer service process thanks to the usage of predictive tools such as the one presented in this work, an increase in efficiency up to 90% can be obtained, together with a reduction in the operating costs between 25% and 30%.

Similar machine learning methods have already been used in order to solve resource allocation problems in order to improve the perceived QoS in [27–29]. In more detail, Pietrabissa et al. have already focused on distributed load balancing in Software Defined Networking relying on Lyapunov-based decision-making algorithms in [27], on optimal buffer allocation for guaranteeing QoS in multimedia Internet broadcasting for mobile networks in [28], and on predictor-based control design for improving Quality of Experience in delay-sensitive Future Internet frameworks in [29]. In contrast with the cited works, this paper specifically focuses on the prediction of the ticket reopening phenomena that characterize fixed networks and therefore also 5G fiber optic networks. To this aim, we exploited several machine learning approaches and compared the obtained performance results.

The paper is organized as follows: Section 2 introduces the so-called Analytical Base Table (ABT), namely the dataset to be given as an input to the predictive model for training and test purposes, as well as the data collection and preparation activity that has been carried out to assemble said ABT. Section 3 presents the performance achieved by different machine learning based classifiers, comparing the results obtained on the original dataset against those obtained on the reduced dataset. Section 4 discusses a further dimensionality reduction effort before introducing the Bayesian network classifier whose performance is the best in class. Concluding remarks end the paper.

## 2. Data Collection, Data Preparation and Analytical Base Table

With the aim of effectively predicting ticket reopening relative to the QoS perceived in 5G fiber optic networks—namely, according to the emerging FWA paradigm—the methodological approach inspiring this work can be structured as follows.

- A first *data collection* and *data preparation* effort were made, by collecting, aggregating, cleaning and preparing the relevant data for the subsequent processing phase, resulting in a homogeneous ABT to be fed to the predictive analytics/classification engine that was subsequently designed and developed. This first activity was carried out in the framework of an extensive joint research initiative on Next Generation Networks between ELIS Innovation Hub and Vodafone over the years 2018–2021.
- Then, the *KPI definition* was necessary, that is, the identification of the relevant KPIs that allows for quantifying the benefits that are ultimately yielded by the adoption of the proposed predictive analytics engine. In this respect, we chose to evaluate the performance of the proposed predictive analytics engine in terms of accuracy, Gini coefficient, Youden index, and Area Under the ROC Curve (AUC).
- The design and training of the *predictive analytics engine* followed, together with the evaluation of the performance obtained on a suitable test dataset.

In more detail, the relevant input data are collected from two heterogeneous data sources,

1. the former related to the Virtual Unbundled Local Access (VULA) technology—VULA offers a means to any licensed network operator to effectively join the ultrabroadband network infrastructure of the backbone provider by virtually accessing the last mile only;
2. the latter related to the Sub-Loop Unbundling (SLU) technology—SLU is alternative to VULA, yet it joins the network infrastructure of the backbone provider only at the copper section, thus exhibiting lower performance than VULA.

The *original* dataset  $X$  is therefore represented by an  $m \times n \times t$  matrix aggregating the inputs from both VULA and SLU technologies. After being suitably cleaned, it reports  $n = 307$  network QoS parameters collected over a period of  $t = 30$  days of intensive usage for a group of  $m = 600,000$  users.

The  $i$ -th row of  $X$  (for  $i = 1, \dots, m$ ),

$$x_i = (x_{u,v}^i)_{\substack{u = 1, \dots, n, \\ v = 1, \dots, t}} \quad (1)$$

associated with a user  $i$  that the customer service has already closed a ticket for (since the aim of the paper is to predict ticket *reopening*), accounts for the values of the  $n$  network QoS parameters perceived by user  $i$  on day  $v$ . Among the  $m$  users, in the considered period, 25% reopened a ticket due to a technical issue that is still unsolved even though a ticket in that respect had already been closed. In particular, data collection was carried out with a specific data matching criterion: namely, in the case of a reopened ticket, only the last data point—i.e., the last values of the  $n$  features—that is, the temporally closest to ticket reopening for the considered user is collected and stored in the dataset  $X$ , whereas, for all other tickets—that have already been closed and have not been reopened yet—all data in the time interval between ticket closing and the last possible sampling instant are collected and stored in the dataset  $X$ .

In order to obtain a smaller dataset, with reduced dimensionality ( $p < n = 307$ ) but with very similar information content, we first performed feature reduction on the original dataset  $X$  by removing all features with very low variance, i.e., proving to be redundant when it comes to estimating the probability of ticket reopening.

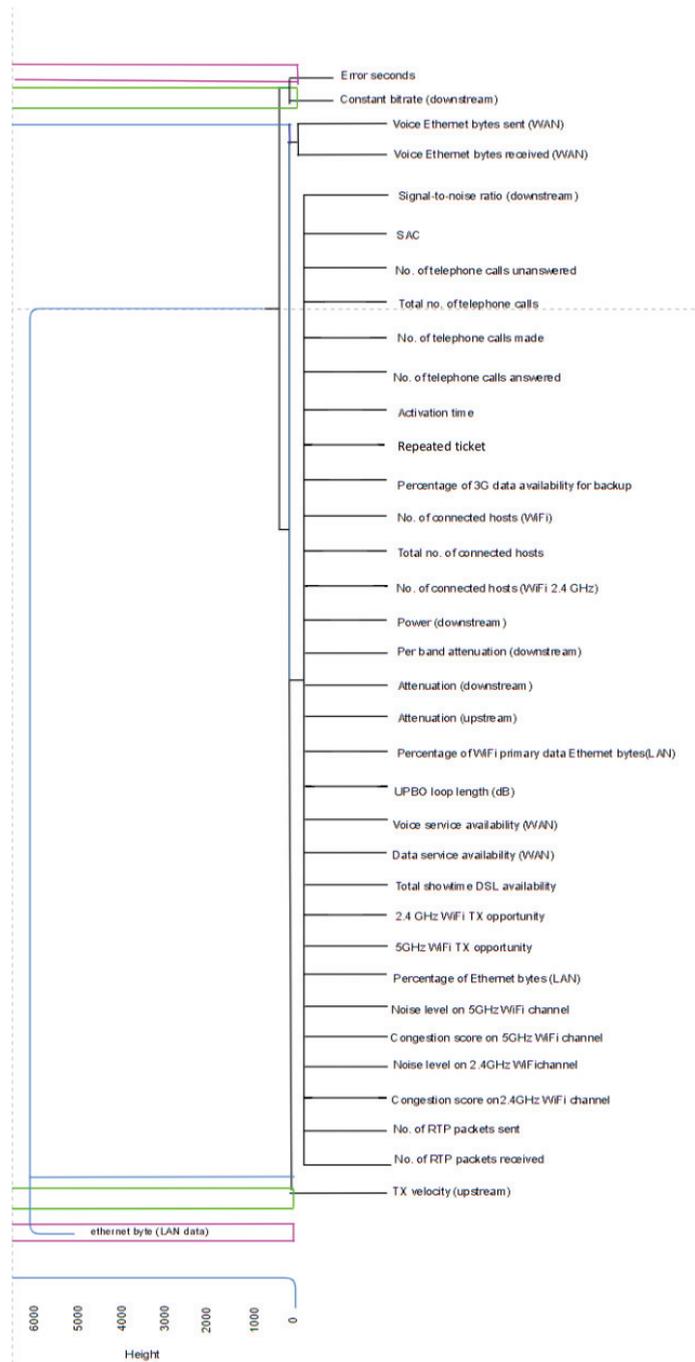
Then, we carried out a linear correlation analysis among the features which, however, did not yield any relevant results; this is why we chose to resort to *hierarchical clustering* in order to shed light on any existing nonlinear correlations.

The result of hierarchical clustering, performed on the portion of  $X$  accounting for the VULA and SLU inputs alternatively, is represented by the *dendrograms* shown in Figures 1 and 2, which represent the resulting hierarchies of clusters, by reporting:

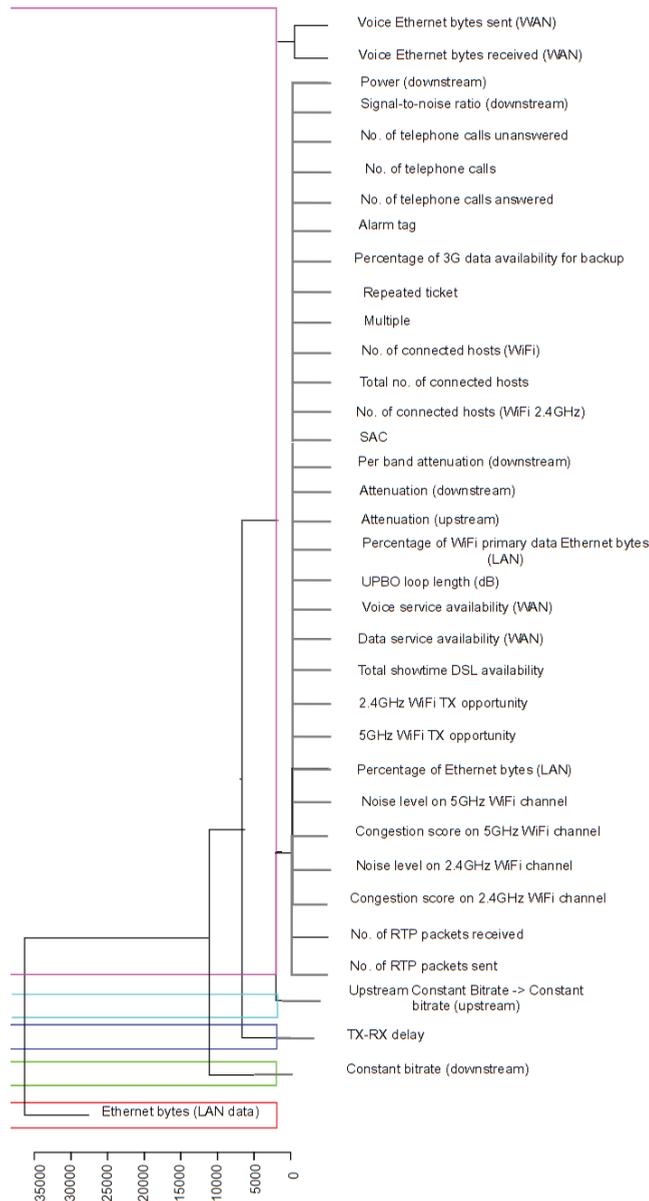
- along the  $x$ -axis the logical distance between clusters according to the Manhattan metric;
- along the  $y$ -axis the hierarchical level of aggregation (denoted with 'Height' in Figures 1 and 2) measured in integer positive values.

In both dendrograms, important correlations emerged—which had not been as evident from linear correlation analysis at all—between ticket reopening (accounted for by the 'Repeated ticket' variable) and some network parameters. The most relevant evidence is related to the correlations of the Repeated ticket variable with the following:

- Percentage of 3G data availability for backup;
- Alarm tag;
- Number of connected hosts (WiFi);
- Total number of connected hosts.



**Figure 1.** Dendrogram showing the results of hierarchical clustering on the VULA portion of the original dataset X.



**Figure 2.** Dendrogram showing the result of hierarchical clustering on the SLU portion of the original dataset X.

As regards the last two, it is reasonable that the higher the number of connected devices, the higher the performance degradation.

A specific remark has to be made relative to the tight correlation of the Repeated ticket variable with the Alarm tag. Indeed, the Alarm tag parameter is a boolean variable which is valued 1 if the so-called customer-premises equipment parameter hits the alarm and 0 otherwise according to two alarm-triggering rules:

- if in almost 4 out of 8 measurements, the SAC parameter goes below threshold;
- if in 3 consecutive measurements, the SAC parameter goes below the threshold.

The SAC variable, which specifically accounts for attenuation-related QoS, can have values in the range from 1 to 6 (where 1 accounts for ‘bad’ and 6 for ‘good’) and is obtained by comparing other two parameters, namely signal-to-noise ratio (downstream) and attenuation (downstream), respectively.

The correlation between Repeated ticket and Alarm tag can be considered as *primary* since it is the tightest one. Therefore, in addition to the primary correlations highlighted by the dendrograms, starting from the tight correlation between Repeated ticket and Alarm tag, other *secondary* correlations also emerge between the Repeated ticket variable, on the one hand, and SAC, signal-to-noise ratio (downstream) and attenuation (downstream), on the other hand. In particular, by observing the relationship among these variables, we can notice the following high degrees of correlation:

- $-47\%$  correlation between Alarm tag and SAC for both VULA and SLU technologies;
- $+65\%$  correlation between SAC and signal-to-noise ratio for the SLU technology and  $+68\%$  correlation between the same variables for the VULA technology;
- $-55\%$  correlation between SAC and attenuation (downstream) for the SLU technology and  $-45\%$  between the same variables for the VULA technology.

It is also possible to evaluate the similarity degree between the two dendrograms by computing the so-called *entanglement* parameter—ranging from 0, which accounts for no entanglement—to 1—which accounts for full entanglement. A low entanglement score implies a good alignment degree between the QoS performance of the two technologies with respect to the time period and set of users observed. In the considered case, the entanglement parameter is evaluated to be equal to 0.175, thus allowing us to consider as ABT a reduced dataset  $X^{red}$  extracted from the original one  $X$ . In particular, the Repeated ticket variable occupies very similar positions in both dendrograms. In light of this, it is reasonable to adopt the same predictive algorithm to predict ticket reopening for both VULA and SLU technologies.

As a result, the following lessons can be learned from the data preparation activity carried out in this section:

- no relevant linear correlations emerge between the Repeated ticket variable and QoS parameters;
- the variables exhibiting primary nonlinear correlations with the Repeated ticket variable are the percentage of 3G data availability for backup and the Alarm tag;
- the variables exhibiting secondary nonlinear correlations with the Repeated ticket variable are SAC, signal-to-noise ratio, and attenuation in downstream;
- the datasets related to the VULA and SLU technologies are very much correlated with each other due to the 0.175 entanglement coefficient computed between the two dendrograms in Figures 1 and 2.

Hence, the *reduced* dataset  $X_{red}$  we are going to resort to hereinafter can be regarded as a  $p \times n \times t$  matrix reporting the following  $p = 15$  network QoS parameters for the same group of  $m$  users at time  $t$ : (i) SAC, (ii) signal-to-noise ratio (downstream), (iii) attenuation (downstream), (iv) constant bitrate (downstream), (v) downstream maximum rate, (vi) Ethernet bytes (LAN data), (vii) percentage of Ethernet bytes (LAN), (viii) percentage of WiFi primary data Ethernet bytes (LAN), (ix) attenuation (upstream), (x) current bitrate (upstream), (xi) upstream maximum rate, (xii) power (upstream), (xiii) UPBO (upstream power back-off) loop length (dB), (xiv) ticket close code, and (xv) repeated ticket.

The ticket close code has not been mentioned so far but was already also present in the original dataset  $X$ : it testifies that the network operator has acknowledged the causes that are attributable to the variations in the QoS parameters for which the user is complaining. The ticket close code variable may take one of the following values: (a) activity on the OCA protocol, (b) existence of a known problem that is pending for resolution, (c) line/device problem needing for device reboot, (d) line/device problem

needing for device reset, (e) unexploited link/GNP (Geographic Number Portability), (f) LNI-minimum bitrate, (g) maximum obtainable performance, (h) no trouble found, (i) OLO2OLO problem—OLO2OLO is the Italian platform allowing the migration of access lines between different network operators insisting on the same network infrastructure—, (j) performance degradation resulting from monitoring, (k) macroproblem due to the network infrastructure, (l) known network outage, (m) network outage detected as a result of monitoring, (n) access degradation of the network infrastructure resulting from monitoring, (o) access network degradation relative to the backbone provider, (p) access network degradation relative to the network operator, and (q) wrong assignment of the network service to a user. For the sake of completeness, in Table 1, we show the relative frequency of each ticket close code in the considered dataset.

**Table 1.** Relative frequencies of the values exhibited by the ticket close code variable.

Ticket Close Code	Relative Frequency
Known network outage	42.99%
Network outage—monitoring	22.35%
Access network degradation—backbone provider	13.14%
Network infrastructure macroproblem	12.48%
No trouble found	4.10%
Line/device problem—reset device	1.74%
LNI-minimum bitrate	1.71%
Known problem pending for resolution	0.44%
OLO2OLO problem	0.41%
Activity on the OCA protocol	0.30%
Access network degradation—network operator	0.10%
Maximum obtainable performance	0.10%
Network infrastructure access degradation—monitoring	0.04%
Performance degradation—monitoring	0.04%
Unexploited link/GNP	0.03%
Line/device problem—reboot device	0.02%
Wrong assignment	0.02%

We now present our predictive model. In more detail, we are going to address the following classification problem: *is a user at risk of reopening a ticket that was previously closed even though the related technical issue (in terms of perceived QoS) was still not solved?*

### 3. Classification for Predicting Ticket Reopening

In this section, we show the performance achieved by different machine learning based classifiers trained both on the original dataset  $X$  and on the reduced dataset  $X_{red}$  in order to predict if a monitored ticket will be reopened or not. This allows us to evaluate the effectiveness of the dimensionality reduction activity discussed in the previous section.

#### 3.1. Different Approaches to Classification

Machine learning is a branch of artificial intelligence based on the idea that systems can learn from data and make reasonable decisions with minimal human intervention. In contrast with many statistical modeling approaches, which generally value inference over prediction, the focus of machine learning is predictive accuracy (see [30]). High predictive accuracy is usually achieved by training complex predictive models, often involving advanced numerical optimization routines, on a very large number of training examples.

According to the survey provided in [31], in this paper, the following supervised classification techniques are considered: decision tree, random forest, boosting, logistic regression, Artificial Neural Network (ANN) and SVM.

The chosen architecture for the decision tree based classifier is inspired by [32]. The random forest based approach to classification follows [33]. The setup for boosting resembles [34], whereas the logistic regression one is inspired by [35].

Instead, in the case of the ANN, we consider a two-layer fully connected network. For the hidden layer, we resort to ReLU nonlinearity, whereas, for the output layer, we have a Softmax loss function. The size of the neural network for the input and output layers is dependent on the input dataset ( $X$  and  $X_{red}$ , alternatively) and classes respectively, while the hidden layer is arbitrarily set.

Finally, the SVM classifier follows the classical approach from [36].

### 3.2. Numerical Simulations and Results

In this subsection, we compare the different classification approaches (described in Section 3.1) in order to predict ticket reopening via supervised learning. According to Section 2, we consider both datasets: the first one ( $X$ ) in the original form and the second one in the reduced form  $X_{red}$  (through feature selection analysis).

Given a month (i.e., 30 days) of data collected according to the format discussed in Section 2, we reordered the dataset by picking six groups of four days as training sets ( $k \in \{1, 6, 11, 16, 21, 26\}$ ), denoting them, with a slight abuse of notation, with  $X_{training}[k]$  in the case of the original dataset and with  $X_{training}^{red}[k]$  in the case of the reduced dataset, that is,

$$X_{training}[k] := \begin{bmatrix} x_{u,(v=k)}^i \\ x_{u,(v=k+1)}^i \\ x_{u,(v=k+2)}^i \\ x_{u,(v=k+3)}^i \end{bmatrix}, \quad \forall u, \quad k \in \{1, 6, 11, 16, 21, 26\}, \quad i = 1, \dots, m, \quad (2)$$

and analogously for  $X_{training}^{red}[k]$ .

We then considered the dataset portion relative to each of the remaining six days of the considered month ( $q = k + 4$ ) as a one-day subset of the ABT providing a suitable test set, denoted with

$$X_{test}[q] := [x_{u,(v=q)}^i], \quad \forall u, \quad k \in \{1, 6, 11, 16, 21, 26\}, \quad i = 1, \dots, m, \quad (3)$$

in the case of the original dataset, and with  $X_{test}^{red}[q]$  defined analogously, in the case of the reduced dataset.

In order to ensure the statistical robustness of the learned models, we proceeded in the following way. We first trained each classification algorithm on  $X_{training}[k]$  and  $X_{training}^{red}[k]$  alternatively, in order to obtain the learned models for each iteration  $k$  (*training* phase). Then, we tested each model learned at iteration  $k$  on the one-day test set  $X_{test}[q]$  in the case of the original dataset, and on the one-day test set  $X_{test}^{red}[q]$  in the case of the reduced dataset (*test* phase).

Eventually, we measured the KPIs listed below for each couple  $(k, q)$  of training and test sets and we reported in Tables 2 and 3 the average KPI values over all  $(k, q)$  couples.

For both data preparation and supervised learning algorithms, all codes are written in R. All the simulation runs were performed on a dual-core Intel Core i7-7500U 2.70GHz (up to 3.50 GHz) processor equipped with 16 GB RAM and running Ubuntu 18.04.

Numerical results are provided in terms of *Accuracy*, *Gini coefficient*, *Youden index* and *AUC*.

- *Accuracy* [37]: the accuracy measure tells how well a machine learner, which learned the hypothesis  $h$  as the approximation of the target classification function  $V$ , performs in terms of classifying a novel unseen example correctly. The true error of hypothesis  $h$  is the probability that it will misclassify a randomly drawn example  $x$ , that is,

$$error(h) = Pr[V(x) \neq h(x)]. \quad (4)$$

With this in mind, accuracy has the following definition:

$$Accuracy = \frac{\text{number of correct predictions}}{\text{total number of predictions}} \tag{5}$$

For binary classification, as in the considered case, accuracy can also be calculated in terms of positives and negatives as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

where *TP*, *TN*, *FP* and *FN* are the number of true positives, true negatives, false positives and false negatives, respectively.

- *Gini coefficient* [37]: It provides a measure of the degree or probability of the target variable being wrongly classified. It has values between  $-1$  and  $1$ . The closer it is to  $1$  the better.
- *Youden index* [37]: it is a goodness-of-fit index that represents the maximum separation between the model Receiver Operating Characteristic (ROC) curve and the baseline ROC curve.
- *AUC* [37]: an ROC curve shows the probability distribution just for the events, the positive class. It compares the events predicted correctly (true positives) against the events predicted incorrectly (false positives). The higher the sensitivity (true positive rate close to  $1$ ) and the lower the false negative rate (specificity close to  $1$ ), the better the model. The AUC gives us the area under the ROC curve: the greater the area, the higher the index.

**Table 2.** Numerical results for the dataset *X*.

Classifier Trained on the Original Dataset <i>X</i>	Accuracy	Gini Coefficient	Youden Index	AUC
Decision Tree	69%	0	0.18	0.5
Random Forest	75%	0.86	0.80	0.93
Gradient Boosting	71%	0.79	0.79	0.89
Logistic Regression	70%	0	0.18	0.5
ANN	66%	0.05	0.06	0.53
Support Vector Machine	73%	0.79	0.79	0.89

**Table 3.** Numerical results for the dataset *X<sub>red</sub>*.

Classifier Trained on the Original Dataset <i>X<sup>red</sup></i>	Accuracy	Gini Coefficient	Youden Index	AUC
Decision Tree	79%	0.1	0.20	0.55
Random Forest	80%	0.9	0.83	0.95
Gradient Boosting	79%	0.8	0.84	0.90
Logistic Regression	81%	0.12	0.25	0.56
ANN	67%	$-0.12$	0.06	0.46
Support Vector Machine	69%	0.78	0.78	0.89

The best one in the first case is the random forest algorithm. In the second case, the most accurate is the logistic regression algorithm, but the best performing one in general remains the random forest algorithm.

#### 4. A Bayesian Network Classifier Trained on a Further Reduced Dataset

We now propose another data-driven classification model, namely based on a Bayesian network, aimed at improving the performance already obtained on the dataset *X<sub>red</sub>* by means of a further dimensionality reduction, namely resorting to the further reduced dataset *X'<sub>red</sub>*.

#### 4.1. Bayesian Network Classifier

Based on the reduced dataset, we trained a classifier resorting to a Bayesian network.

A Bayesian network is a probabilistic graphical model that, by representing a set of variables and their conditional dependencies via a directed acyclic graph  $\mathcal{G}$ , allows for predicting the likelihood that one of several possible known causes is the contributing factor behind the occurrence of a specific event. In the considered case, the aim is that of predicting if a combination of network QoS parameters belongs to the discrete class variable Repeated ticket.

In more detail, we learned a Naive Bayes network structure  $\mathcal{G}$  as in Figure 3, revolving around the following input variables, which therefore compose the new reduced dataset  $X'_{red}$  as a  $p' \times n$  matrix, with  $p' = 7$ :

- Repeated ticket;
- Ticket close code;
- Signal-to-noise ratio (downstream);
- SAC;
- Constant bitrate (downstream);
- Power (upstream);
- Attenuation (upstream).

We set the Upstream Attenuation node as root node and the SAC and Repeated ticket nodes as leaf nodes.

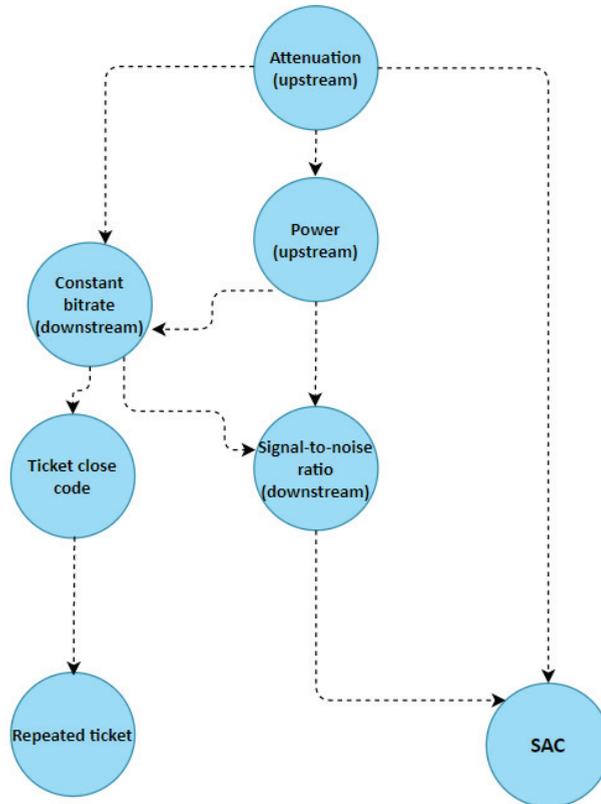


Figure 3. Naive Bayes network structure  $\mathcal{G}$  behind the chosen Bayesian network classifier.

The variables’ alarm tag and percentage of 3G data availability for backup have been preliminarily excluded from the ABT because, from a statistical viewpoint, they were not suitable for the algorithm procedure that is behind training a Bayesian network classifier.

In addition, before training the classifier, we chose to perform *discretization*—referred to as the process of grouping values into intervals in order to limit the number of possible states—on the input data according to the type and distribution of each variable, in order to optimize the performance in the creation of the Bayesian network graph. The following two methods were applied onto the continuous variables of the ABT: namely, quantile (subdivision by frequency) and uniform (subdivision into a suitable number of groups of the same size) discretization.

Quantile discretization was performed onto the constant bitrate (downstream), signal-to-noise ratio (downstream), attenuation (upstream) and power (upstream) variables, by grouping the values of each variable into four same size bins, split based on percentiles.

Uniform discretization, instead, was performed onto the ticket close code and SAC, grouping the values of each variable into four same-width discrete bins depending on the span of possible values for each considered variable.

The Repeated ticket variable was discretized into two disjoint bins, of which 15% are repeated tickets and the rest are non-repeated.

Table 4 shows the characteristics of the Bayesian network in detail. The value of the Pearson correlation coefficient (denoted with ‘Strength’ in the table) indicates the existing degree of correlation between the variables considered: the closer this value is to 1, the greater the correlation between the variables. On the other hand, the ‘Direction’ column indicates the degree of reliability of the links that introduce a hierarchy between the variables: in this case, too, the closer the value is to 1, the more the direction of the link accounting for the existing relationship between the considered variables is reliable.

**Table 4.** Pearson correlation coefficient between the variables in  $X'_{red}$  and degree of reliability of the links characterizing the learned Naive Bayes network structure  $\mathcal{G}$ .

From	To	Strength	Direction
Signal-to-noise ratio (downstream)	SAC	1.0	0.93
Power (upstream)	Signal-to-noise ratio (downstream)	1.0	0.9
Attenuation (upstream)	SAC	1.0	1.0
Attenuation (upstream)	Power (upstream)	1.0	0.7
Power (upstream)	Signal-to-noise ratio (downstream)	1.0	0.6
Attenuation (upstream)	Constant bitrate (downstream)	1.0	1.0
Power (upstream)	Constant bitrate (downstream)	0.87	0.85
Constant bitrate (downstream)	Ticket close code	0.93	0.9
Ticket close code	Repeated ticket	0.93	0.95

It is clear from Figure 3 and Table 4 that the correlation coefficient with the variables closest to the Repeated ticket variable is always greater than 0.93, which implies that the proposed tree structure can be considered as highly reliable for our classification purpose.

The combinations of conditional probabilities calculated by the Bayesian network as a result of discretization generate a number of scenarios to which it is possible to associate the probability of occurrence of the event of ticket reopening. For the considered reduced dataset  $X'_{red}$ , more than one thousand different simulation scenarios were generated. Namely, the 13 intervals shown in Table 5 are combined with the 17 possible causes identified within the ticket close code variable. Constant bitrate in downstream is measured in bits per second. Attenuation in upstream is measured in dB and power in upstream is measured in dBmV.

**Table 5.** Relevant distributions of the variables of  $X'_{red}$  into suitable bins as a result of discretization.

Variable	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5
Constant bitrate (downstream) [bits per sec]	<36,200	[36,200; 54,000]	[54,000; 64,200]	[64,200; 87,100]	>87,100
Attenuation (upstream) [dB]	<10.2	[10.2; 16.1]	[16.1; 21.1]	>21.1	-
Power (upstream)	<-19.2	[-19.2; -8.6]	[-8.6; -0.7]	>0.7	-

**Remark 1.** The number of relevant scenarios may vary depending on the discretization type and the number of tickets associated with the different combinations. This has proven to be the best choice, given the characteristics of the considered dataset.

4.2. Model Performance Evaluation and Discussion on the Results

For the purpose of evaluating the model performance, we adopted the same approach as discussed at the beginning of Section 3.2. However, in order to further test the robustness of the Bayesian network classifier, we also carried out the experiment of creating 1000 random pairs of training/test sets according to the 70/30 rule, i.e., 70% of the  $X'_{red}$  dataset was used for training purposes and the rest for testing. The measured KPIs were very similar, thus testifying the effectiveness of the Bayesian network classifier as well as its robustness. The average values of the KPIs measured throughout these tests are reported below in Table 6 in order to compare them with the results of the different classifiers trained in Section 3.

The validation process was completed by comparing the accuracy measure achieved by the Bayesian network classifier against the performance of the other classifiers.

From Table 6, it is clear that the Bayesian Network classifier trained on  $X'_{red}$  outperforms the classifiers introduced in Section 3.

**Table 6.** Numerical results for the second dataset.

Trained Classifier	Accuracy	Gini Coefficient	Youden Index	AUC
Decision Tree trained on $X_{red}$	79%	0.1	0.2	0.55
Random Forest trained on $X_{red}$	80%	0.9	0.83	0.95
Gradient Boosting trained on $X_{red}$	79%	0.8	0.84	0.90
Logistic Regression trained on $X_{red}$	81%	0.12	0.25	0.56
ANN trained on $X_{red}$	67%	-0.12	0.06	0.46
Support Vector Machine trained on $X_{red}$	69%	0.78	0.78	0.89
Bayesian Network classifier trained on $X'_{red}$	96%	0.90	0.79	0.95

Among the trained classifiers, according to the accuracy and AUC measures, the Bayesian Network classifier proves to be the most effective at minimizing the error (8).

From the results obtained, we also infer the combinations of features in  $X'_{red}$  that are most probably the reason for ticket reopening; namely, they are the *events* listed below:

- (1) Ticket close code (d) AND Constant bitrate (downstream) in bin 2 AND Power (upstream) in bin 4 AND Attenuation (upstream) in bin 3;
- (2) Ticket close code (k) AND Constant bitrate (downstream) in bin 1 AND Power (upstream) in bin 1 AND Attenuation (downstream) in bin 3;
- (3) Ticket close code (i) AND Constant bitrate (downstream) in bin 4 AND Power (upstream) in bin 3 AND Attenuation (upstream) in bin 3;
- (4) Ticket close code (i) AND Constant bitrate (downstream) in bin 3 AND Power (upstream) in bin 3 AND Attenuation (upstream) in bin 4;
- (5) Ticket close code (i) AND Constant bitrate (downstream) in bin 2 AND Power (upstream) in bin 2 AND Attenuation (upstream) in bin 3;
- (6) Ticket close code (i) AND Constant bitrate (downstream) in bin 3 AND Power (upstream) in bin 2 AND Attenuation (upstream) in bin 1;
- (7) Ticket close code (p) AND Constant bitrate (downstream) in bin 2 AND Power (upstream) in bin 3 AND Attenuation (upstream) in bin 3;

- (8) Ticket close code (p) AND Constant bitrate (downstream) in bin 3 and Power (upstream) in bin 4 AND Attenuation (upstream) in bin 4;
- (9) Ticket close code (p) AND Constant bitrate (downstream) in bin 4 AND Power (upstream) in bin 1 AND Attenuation (upstream) in bin 2.

Table 7 reports them with the corresponding number of occurrences of such combinations of QoS parameters.

**Table 7.** Combinations of features in  $X'_{red}$  that are most probably the reason for ticket reopening according to the predictions of the Bayesian Network classifier.

Event	Number of Occurrences in $X'_{red}$	Percentage of Events Causing a Repeated Ticket
Event (1)	7	42.86%
Event (2)	29	34.48%
Event (3)	17	29.41%
Event (4)	107	20.56%
Event (5)	102	17.65%
Event (6)	102	12.75%
Event (7)	372	12.37%
Event (8)	447	10.96%
Event (9)	504	10.12%

In general, as can be seen from Table 7, the trained classifier provides the customer service of a network operator with a reliable tool for effectively monitoring customer tickets that, despite being already closed, are at risk of being reopened due to unsolved technical issues related to the perceived QoS.

The complexity of the Bayesian Network classifier, namely the most successful one, is linear in the number of training examples and in the number of features characterizing each training example. Instead, almost all other methods exhibit increased runtime complexity: more precisely, the Decision Tree, Random Forest and Gradient Boosting approaches are such that their complexity is logarithmic in the number of training examples, whereas the complexity of the SVM approach is quadratic in the number of training examples. Only the ANN and the Logistic Regression techniques have comparable computational complexity with respect to the Bayesian Network classifier, but with lower predictive performance (as shown in Table 6).

### 5. Conclusions

The paper proposes a data-driven approach based on machine learning for predicting technical ticket reopening in customer service platforms of telecommunications companies providing 5G fiber optic networks, namely with respect to ensuring that, between end user and service provider, the Service Level Agreement in terms of perceived Quality of Service is satisfied.

The activity was carried out within the framework of an extensive joint research initiative on Next Generation Networks between ELIS Innovation Hub and a major network service provider in Italy over the years 2018–2021.

The authors compare the performance of different approaches to classification—ranging from decision trees to Artificial Neural Networks and Support Vector Machines—and establish that a Bayesian network classifier is the most accurate at predicting whether a monitored ticket will be reopened or not.

In addition, the authors propose a suitable dimensionality reduction strategy that proves to be successful at increasing the computational efficiency by reducing the size of the relevant training dataset by two orders of magnitude with respect to the original dataset.

Numerical simulations show the effectiveness of the proposed approach, proving it can be a very useful tool for service providers in order to identify the customers that are most at risk of reopening a ticket due to an unsolved technical issue.

As future work, the authors look forward to testing the proposed method on Quality of Service datasets coming from additional sources and/or related to other 5G networks, as well as to testing the same method on even larger datasets in orders to further assess its scalability properties.

**Author Contributions:** Investigation, M.V.; Methodology, L.R.C., A.C., M.D., E.S. and M.V.; Software, A.C., M.D. and E.S.; Writing—original draft, L.R.C. and N.A.S.; Writing—review & editing, L.R.C. and N.A.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by ELIS Innovation Hub within a collaboration with Vodafone, grant number Joint Research Project within the framework of the Mindset Revolution Semester.

**Data Availability Statement:** Not Applicable, the study does not report any data.

**Conflicts of Interest:** The work presented in this paper was carried out while Caliciotti and Scocchi were with ELIS Innovation Hub and does not reflect the results of any activity carried out at Enel Green Power S.p.A. and ERG S.p.A., to which these two authors are currently affiliated, respectively.

## References

1. Cater-Steel, A.; Valverde, R.; Shrestha, A.; Toleman, M. Decision support systems for IT service management. *Int. J. Inf. Decis. Sci.* **2016**, *8*, 284–304.
2. Keller, A.; Midboe, T. Implementing a service desk: A practitioner’s perspective. In Proceedings of the 2010 IEEE Network Operations and Management Symposium—NOMS 2010, Osaka, Japan, 19–23 April 2010; pp. 685–696.
3. Cusick, J.J.; Ma, G. Creating an ITIL inspired Incident Management approach: Roots, response, and results. In Proceedings of the 2010 IEEE/IFIP Network Operations and Management Symposium Workshops, Osaka, Japan, 19–23 April 2010; pp. 142–148.
4. Tang, X.; Todo, Y. A Study of Service Desk Setup in Implementing IT Service Management in Enterprises. *Technol. Investig.* **2013**, *190*–196. [CrossRef]
5. Tanovic, A.; Mastorakis, N. Advantage of Using Service Desk Management Systems in Real Organizations. *Int. J. Econ. Manag. Syst.* **2016**, *1*, 81–86.
6. Tanovic, A.; Butkovic, A.; Orucevic, F.; Mastorakis, N.E. Advantages of the implementation of Service Desk based on ITIL framework in telecommunication industry. In *Recent Researches in Electrical Engineering*; 2014; pp. 165–179. Available online: <http://www.wseas.us/e-library/conferences/2014/Lisbon/ELEL/ELEL-20.pdf> (accessed on 3 October 2021).
7. Harcenko, M.; Dorogovs, P.; Romanovs, A. IT Service Desk Implementation Solutions. *J. Riga Tech. Univ.* **2010**, *44*, 68–73. [CrossRef]
8. Jäntti, M. *Examining Challenges in IT Service Desk System and Processes: A Case Study*; 2012. Available online: <https://www.semanticscholar.org/paper/Examining-Challenges-in-IT-Service-Desk-System-and-J%3%A4ntti/a0e6be989f664445e5a83535d5dd6b1427dc5e2a> (accessed on 3 October 2021).
9. Serbest, S.; Goksen, Y.; Dogan, O.; Tokdemir, A. Design and implementation of help desk system on the effective focus of information system. *Procedia Econ. Financ.* **2015**, *33*, 461–467. [CrossRef]
10. Robles, V.D. Resolving discourse at technical-support helpdesks. *IEEE Trans. Prof. Commun.* **2018**, *61*, 275–294. [CrossRef]
11. Shae, Z.; Bergstrom, T.; Pinhanez, C.; Podlaseck, M. Multimedia Chat for Helpdesks: A Practical SOA Architecture. In Proceedings of the IEEE Congress on Services—Part I, Honolulu, HI, USA, 6–11 July 2008; pp. 75–77.
12. Zhu, X.; Zhou, W.; Li, T. A visual tool for ticket monitoring and management. In Proceedings of the 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Nanjing, China, 24–26 November 2017; pp. 1–7.
13. Ilieva, R.; Anguelov, K.; Gashurova, D. Monitoring and optimization of e-Services in IT Service Desk Systems. In Proceedings of the 19th International Symposium on Electrical Apparatus and Technologies (SIELA), Bourgas, Bulgaria, 29 May–1 June 2016; pp. 1–4.
14. Zhou, Y.; Tong, Y.; Gu, R.; Gall, H. Combining text mining and data mining for bug report classification. *J. Softw. Evol. Process.* **2016**, *28*, 150–176. [CrossRef]
15. Silva, S.; Pereira, R.; Ribeiro, R. Machine learning in incident categorization automation. In Proceedings of the 13th Iberian Conference on Information Systems and Technologies (CISTI), Cáceres, Spain, 13–16 June 2018; pp. 1–6.
16. Agarwal, S.; Sindhgatta, R.; Sengupta, B. SmartDispatch: Enabling Efficient Ticket Dispatch in an IT Service Environment. In Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 12–16 August 2012; pp. 1393–1401.
17. Wongsakthaworn, R.; Limpiyakorn, Y. Development of IT Helpdesk with Microservices. In Proceedings of the 8th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 15–17 June 2018; pp. 31–34.
18. Aggarwal, V.; Agarwal, S.; Dasgupta, G.B.; Sridhara, G.; Vijay, E.V. ReAct: A System for Recommending Actions for Rapid Resolution of IT Service Incidents. In Proceedings of the IEEE International Conference on Services Computing (SCC), San Francisco, CA, USA, 27 June–2 July 2016; pp. 1–8.

19. Son, G.; Hazlewood, V.; Peterson, G.D. On Automating XSEDE User Ticket Classification. In Proceedings of the 2014 Annual Conference on Extreme Science and Engineering Discovery Environment, Atlanta, GA, USA, 13 July 2014; pp. 1–7.
20. Wang, D.; Li, T.; Zhu, S.; Gong, Y. iHelp: An Intelligent Online Helpdesk System. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **2010**, *41*, 173–182. [[CrossRef](#)] [[PubMed](#)]
21. Wang, X.; Zhang, L.; Xie, T.; Anvik, J.; Sun, J. An Approach to Detecting Duplicate Bug Reports Using Natural Language and Execution Information. In Proceedings of the 30th International Conference on Software Engineering, Virtual, 11–13 October 2021; pp. 461–470.
22. Rajeshwari, M.R.; Kavitha, K.S. Reduction of duplicate bugs and classification of bugs using a text mining and contingency approach. *Int. J. Appl. Eng. Res.* **2017**, *12*, 2840–2847.
23. Available online: <https://freshdesk.com/customer-service-training/customer-service-metric> (accessed on 1 September 2021).
24. Skubic, B.; Fiorani, M.; Tombaz, S.; Furuskar, A.; Martensson, J.; Monti, P. Optical transport solutions for 5G fixed wireless access. *IEEE/OSA J. Opt. Commun. Netw.* **2017**, *9*, 10–18. [[CrossRef](#)]
25. IBM Case Studies Corporate Landing Page, AI for Customer Service. Available online: <https://www.ibm.com/case-studies/> (accessed on 4 October 2021).
26. Daugherty, P.R.; Wilson, H.J. *Human+Machine: Reimagining Work in the Age of AI*; Harvard Business Review Press: Brighton, MA, USA, 2018.
27. Pietrabissa, A.; Celsi, L.R.; Cimorelli, F.; Suraci, V.; Priscoli, F.D.; Di Giorgio, A.; Giuseppe, A.; Monaco, S. Lyapunov-Based Design of a Distributed Wardrop Load-Balancing Algorithm With Application to Software-Defined Networking. *IEEE Trans. Control Syst. Technol.* **2018**, *27*, 1924–1936. [[CrossRef](#)]
28. Caliciotti, A.; Celsi, L.R. On optimal buffer allocation for guaranteeing quality of service in multimedia internet broadcasting for mobile networks. *Int. J. Control Autom. Syst.* **2020**, *18*, 3043–3050. [[CrossRef](#)]
29. Celsi, L.R.; Bonghi, R.; Monaco, S.; Normand-Cyrot, D. Normand-Cyrot. On the Exact Steering of Finite Sampled Nonlinear Dynamics with Input Delays. *IFAC-PapersOnLine* **2015**, *48*, 674–679. [[CrossRef](#)]
30. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
31. Narayanan, U.; Unnikrishnan, A.; Paul, V.; Joseph, S. A survey on various supervised classification algorithms. In Proceedings of the International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 1–2 August 2017; pp. 2118–2124.
32. Ben-Gal, I.; Trister, C. Parallel Construction of Decision Trees with Consistently Non Increasing Expected Number of Tests. *Appl. Stoch. Models Bus. Ind.* **2015**, *31*, 64–78. [[CrossRef](#)]
33. Prinzie, A.; van den Poel, D. Random Forests for multiclass classification: Random MultiNomial Logit. *Expert Syst. Appl.* **2008**, *34*, 1721–1732. [[CrossRef](#)]
34. Elias, C.; Calapez, A.; Almeida, S.; Chessman, B.; Simoes, N.; Feio, M. Predicting reference conditions for river bioassessment by incorporating boosted trees in the environmental filters method. *Ecol. Indic.* **2016**, *69*, 239–251. [[CrossRef](#)]
35. Al-Hawari, F.; Barham, H. A machine learning based help desk system for IT service management. *J. King Saud Univ.—Comput. Inf. Sci.* **2019**, *33*, 702–718. [[CrossRef](#)]
36. Cristianini, N.; Ricci, E. Support Vector Machines. In *Encyclopedia of Algorithms*; Kao, M.Y., Ed.; Springer: Boston, MA, USA, 2008.
37. Powers, D. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation. *J. Mach. Learn. Technol.* **2011**, *2*, 37–63.



Review

# An Analysis on Contemporary MAC Layer Protocols in Vehicular Networks: State-of-the-Art and Future Directions

Lopamudra Hota <sup>1</sup>, Biraja Prasad Nayak <sup>1</sup>, Arun Kumar <sup>1</sup>, G. G. Md. Nawaz Ali <sup>2</sup> and Peter Han Joo Chong <sup>3,\*</sup>

<sup>1</sup> Department of Computer Science & Engineering, National Institute of Technology, Rourkela 769008, India; 519cs1015@nitrkl.ac.in (L.H.); 519cs1005@nitrkl.ac.in (B.P.N.); kumararun@nitrkl.ac.in (A.K.)

<sup>2</sup> Department of Computer Science and Information Systems, Bradley University, Peoria, IL 61625, USA; nali@fsmail.bradley.edu

<sup>3</sup> School of Engineering, Computer and Mathematical Sciences, Auckland University of Technology, Auckland 1010, New Zealand

\* Correspondence: peter.chong@aut.ac.nz

**Abstract:** Traffic density around the globe is increasing on a day-to-day basis, resulting in more accidents, congestion, and pollution. The dynamic vehicular environment induces challenges in designing an efficient and reliable protocol for communication. Timely delivery of safety and non-safety messages is necessary for traffic congestion control and for avoiding road mishaps. For efficient resource sharing and optimized channel utilization, the media access control (MAC) protocol plays a vital role. An efficient MAC protocol design can provide fair channel access and can delay constraint safety message dissemination, improving road safety. This paper reviews the applications, characteristics, and challenges faced in the design of MAC protocols. A classification of the MAC protocol is presented based on contention mechanisms and channel access. The classification based on contention is oriented as contention-based, contention-free, and hybrid, whereas the classification based on channel access is categorized as distributed, centralized, cluster-based, cooperative, token-based, and random access. These are further sub-classified as single-channel and multi-channel, based on the type of channel resources they utilize. This paper gives an analysis of the objectives, mechanisms, advantages/disadvantages, and simulators used in specified protocols. Finally, the paper concludes with a discussion on the future scope and open challenges for improving the MAC protocol design.

**Citation:** Hota, L.; Nayak, B.P.; Kumar, A.; Ali, G.G.M.N.; Chong, P.H.J. An Analysis on Contemporary MAC Layer Protocols in Vehicular Networks: State-of-the-Art and Future Directions. *Future Internet* **2021**, *13*, 287. <https://doi.org/10.3390/fi13110287>

Academic Editors: Michael Mackay and Paolo Bellavista

Received: 25 September 2021

Accepted: 9 November 2021

Published: 17 November 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** MAC classification; vehicular networks; VANETs; distributed MAC; centralized MAC; multi-channel MAC; single-channel MAC; spectrum allocation

## 1. Introduction

In the last few decades, the number of vehicles has witnessed a surge in global transportation systems. Vehicular communication is a pivotal component for intelligent transportation system (ITS) and innovative city development. Rapid urbanization has led to expansion in the utilization of vehicles for transportation, attracting researchers in vehicular ad hoc networks (VANETs). VANETs uses a variety of communication technologies such as short-range wireless LAN (WLAN) and cellular technologies such as long-term evaluation (LTE) and voice over LTE (VoLTE) [1]. Since 1980, VANETs incorporating an ad hoc network framework have grown rapidly, with vehicles interacting via wireless networks. Emerging technologies such as 5G, 6G, cloud edge computing, and SDN have ameliorated VANET communication with the timely delivery of safety and non-safety messages in the recent era.

VANETs have a highly dynamic topology with fast-moving nodes, attracting researchers due to rapid urbanization. The field has drawn the attention of scholars towards research for a safer and more comfortable driving experience in the future. It consists of both inter-vehicular Vehicle-to-Vehicle (V2V) and intra-vehicular Vehicle-to-Infrastructure (V2I) communication [2], where infrastructure is the electronic control units within the

vehicles. Two mechanisms of communication in an ITS are mobile ad hoc networks (MANETs) and VANETs [3]. There is no central administration or fixed infrastructure in MANETs; nodes self-configure themselves and interact. Unlike MANETs, VANETs follows predictable patterns along a road. Moreover, the processing and storage capacity of VANETs is better than that of MANETs. VANETs have unique features such as high mobility and the constraint of a dynamic road network topology, unpredictable network size, and infrastructure support that differentiates it from MANETs. A basic model diagram of VANETs includes vehicles and other infrastructures communicating via V2V, V2I, Road Side Units (RSUs), and Onboard Units (OBUs). These communications play a vital role in the transportation system to improve traffic efficiency and safety. A schematic representation of the communication is depicted in Figure 1.

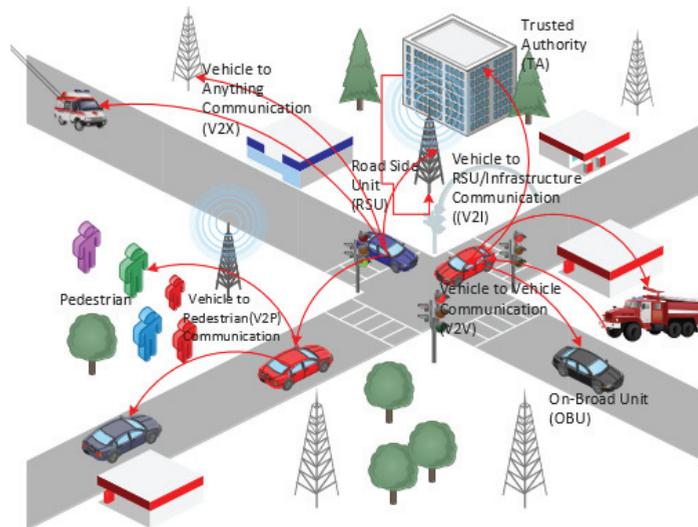


Figure 1. VANET architecture [1].

VANET applications are divided into categories such as life-critical applications, safety, warning applications, e-toll collection, group communication, traffic management, and user applications, as in [4]. Most papers have broadly classified VANET applications into two categories: safety-based applications (handling real-time traffic and avoiding accidents) and non-safety-based applications (infotainment services, parking availability, GPS tracking of nearby places, etc.) [5,6]. The salient characteristics of VANETs include the dynamic topology, unpredictable network size, high scalability, recurrent information exchange, time-critical communication, wireless medium of communication, energy efficiency, and real-time applications [7,8].

Intelligent transportation systems (ITSs) in smart cities have improved road safety and minimized the risk of mishaps on the road. The use of DSRC [9] (IEEE 802.11p [10]), WAVE (IEEE 1609 protocol stack [11]), and 5G LTE have regulated communication in VANETs extensively. Different countries of the world have adopted dedicated frequency bands for vehicular communication by ITS [12]. Based on the division of a spectrum, the MAC protocols are divided into single-channel MAC and multi-channel MAC. The single-channel MAC focuses on resource allocation whereas the multi-channel MAC deals with collision avoidance and load balancing by providing multiple channel access mechanisms.

Some of the challenges tackled by MAC protocol design in VANETs are (1) the provision of low latency safety services and high throughput non-safety services, (2) the elimination of hidden/exposed terminal problems due to rapid mobility of nodes in VANETs and

topology changes, (3) proper resource and bandwidth allocation in single-channel as well as multi-channel for load balancing and better network throughput.

Various MAC protocols have been reviewed in the past based on different criteria. The authors in [13] reviewed multi-channel MAC protocols in VANETs, with the design of an adaptive MAC protocol to handle topology changes by adjusting the control channel based on a Markov model. In [10], TDMA-based MAC protocols were studied, protocols were classified based on topology, and a comparative analysis along with the advantages of using contention-free access mechanism were provided. In [14], the authors assessed foreground multi-layer challenges for better performing VANETs, with a focus on different layers, and proposed solutions along with the limitations and future work.

### 1.1. Paper Contribution

Researchers have expressed considerable interest in vehicular networks, their applications (safety and non-safety), traffic conditions, network topology, protocols, and enhanced network performance. This survey provides the latest update on applications, standards, and MAC protocols for efficient vehicular communication. The state-of-the-art covers the essential aspects of vehicular communications, including architecture, applications, challenges, vehicular networks, standards in different countries, and classification of MAC protocol based on the channel access mechanism. To the best of our knowledge, this survey is the first of its kind to classify MAC protocols based on channel access mechanisms, including the latest protocols. Apart from the classification, the paper presents the details of ITS standard deployments in different regions, followed by a detailed explanation and implementation of the latest-generation technologies (C-V2X, 5G, and SDN). The article commences from a clear description of vehicular communication, architectural overview, objectives, and the state-of-the-art and concludes with the future scope and research directions.

### 1.2. Paper Organization

This paper represents a comprehensive study on MAC protocols in VANETs, the challenges, and issues related to the design of an efficient MAC protocol. The remainder of this paper is organized as follows: Section 2 provides a brief overview of DSRC and cellular technology for vehicular networks. Section 3 introduces some of the traditional MAC protocols along with their pros and cons. Section 4 represents studies on various recently proposed MAC protocols based on their objectives, mechanisms, advantages, and disadvantages. Section 5 discusses the future scope and challenges for designing a MAC protocol. Finally, we conclude this paper in Section 6.

## 2. Vehicular Networks

This section presents DSRC-based networks, their usage and challenges, WAVE architecture, and frequency range standards for vehicular communication of countries such as Europe, Korea, China and Japan. This section also illustrates cellular networks for vehicular communication, including LTE, C-V2X, and 5G.

### 2.1. DSRC-Based Networks

Standard organizations have allocated different frequency bands to different regions for efficient VANETs communication. This frequency spectrum allocation provides multi-channel capabilities with minimal collision and congestion during transmission. In the US, the Federal Communication Commission (FCC) allocated 75 MHz bandwidth for DSRC in the frequency band of 5.9 GHz [15]. Similarly, in Europe, the European Telecommunications Standards Institute (ETSI) provides vehicular communication for V2V and V2I with an allotted range of 50 MHz bandwidth in the frequency band of 5.855 to 5.905 GHz.

On the other hand, China has been provided by the Ministry of Industry and Information Technology (MIIT) with a dedicated 20 MHz and bandwidth range in 5.905–5.925 GHz. In Korea, the Ministry of Science and Information and Communications Technology

(MSIT) allocates a frequency band of 10 MHz for V2I communication with a bandwidth of 5.895–5.905 MHz for CCH and SCH. For Japan, the Association of Radio Industries and Businesses (ARIB) is the organization that allocated 80 MHz as the frequency for DSRC, with bandwidths of 5.770–5.850 GHz and 755.5–764.5 MHz [16].

The WAVE architecture has seven channel divisions, each with 10 MHz, and 5 MHz is kept aside for future use as backup [17]. One of these channel divisions is reserved for safety applications through control channels (CCH), and the rest are for safety and non-safety applications through service channels (SCHs). The transceivers sense these fixed channels for multi-channel access at the same time interval without collision. Orthogonal frequency division multiplexing (OFDM) is used in the WAVE standard for interference avoidance during transmission [18], with data rates of 6–27 Mbps. The ITS-G5 standard uses 2 amplitude-shift keying (2ASK) or 2 phase-shift keying (2PSK), with data rates of 6–12 Mbps; quadrature phase-shift keying (QPSK) supports data rates of 1–4 Mbps. These are some of the modulation techniques used for vehicular communications. Vehicle-to-everything (V2X) communication with LTE and 5G has been adopted along with QPSK [19], which provides a high data rate for better transmission efficiency compared with OFDMA and 2ASK. Table 1 summarizes these standards.

**Table 1.** Vehicular communication standards.

Country	Standards	Frequency Range (GHz)	Data Rates (Mbps)	Modulation
US	WAVE	5.80–5.925	6–27	OFDM
Europe	ITS/G5	5.85–5.905	6–12	2ASK/2PSK
China	C-V2X	5.905–5.925	100	QPSK/16QAM/64QAM/256QAM
Japan	STD/T75	5.770–5.880	1, 4	2ASK/QPSK
Korea	DSRC	5.895–5.905	9, 12, 18, 24, 27	QPSK/BPSK/2ASK

In IEEE 802.11 [20], the distributed coordination function (DCF) deals with medium access based on CSMA with collision avoidance (CSMA/CA). Here, the device first listens to the network channel before transmitting for collision avoidance. In IEEE 802.11, the main focus is on RTS/CTS/ACK mode of packet exchange to access the medium. A network allocation vector (NAV) is set according to the transmission duration indicated by RTS. However, CSMA/CA is not suitable for real-time scenarios due to its inherent channel access delay. The vehicular networks incorporate dedicated short-range communication (DSRC) for enriching the driver's comfort and safety. Wireless access in vehicular environment (WAVE) [21] defines the IEEE 802.11p for MAC layer implementation in VANETs. The DSRC documents the physical (PHY) and medium access control (MAC) layers of the WAVE stack [22]. To achieve quality of service (QoS), the WAVE stack of IEEE 802.11p for a MAC protocol incorporates an enhanced distributed channel access (EDCA) mechanism [23]. The messages are divided into four types based on access priority, as access category AC[0]~AC[3] with separate contention windows and frames set for each category.

## 2.2. Cellular-Based Networks

The Third Generation Partnership Project (3GPP) introduced cellular-V2X (C-V2X) in Release 14 [24]. The use cases of the ITS spectrum in various countries per government standardization is presented in [25,26]. A simulation model of the Third Generation Partnership Project (3GPP) Release 14 Cellular Vehicle-to-Everything (C-V2X) sidelink, upon which 5G New Radio mode 2 was based, is presented in [27]. To support V2X communications, C-V2X Mode 4 modifies the PHY and MAC layers of the LTE sidelink [28]. In high mobility situations, LTE's PHY layer is designed to improve performance. The sensing-based semi-persistent scheduling (SB-SPS) mechanism is implemented at the MAC layer to autonomously select vehicle resources.

In [29], to support beacon broadcasting over V2V networks, a distributed network coding MAC protocol (NC-MAC) is proposed. By combining re-transmission, network coding, and preamble-based feedback mechanisms, reliability is improved. In various situations,

including highway and urban, simulations demonstrate the performance gains obtained from the NC-MAC protocol compared with the 5G cellular vehicle-to-everything (C-V2X) MAC protocol. The DSRC protocol uses a random back-off scheme to propagate data between MAC and PHY, which causes the inter-layer data propagation delay. However, it remains within microseconds. Rate control is therefore directly proportional to end-to-end latency. The C-V2X MAC, however, can introduce a time offset (on average 50 ms) when translating a packet because of semi-persistent scheduling (SPS) operations [30].

As a part of the C-V2X communication, other technologies such as radar, cameras, and in-vehicle sensors support services such as semi-autonomous driving, autonomous driving, and assisted driving in these vehicles. In addition to better coverage and lower deployment costs, V2X services can be run on a dedicated network using the 5G public network, which offers high QoS features with its flexible network design, including ultra-fast data throughput and ultra low latency [31].

Release 14 of C-V2X released in March 2017 included the first version of the tool. The automotive vertical needs were addressed by a specification called LTE-V2X, which was created in conjunction with the development of the 4G LTE system. As a part of the LTE V2X revisions since Release 15, the 3GPP has begun looking at the 5G specification for 5G-V2X. NR V2X specifications defining the sidelink, which are integrated with the 5G-V2X specifications defined in Release 16, in June 2020. The sidelink 5G-V2X standard will be improved in Release 17, which is still undergoing testing and is scheduled for release in 2022 [32].

The framework of 5G is based on the fourth-generation LTE mobile standard. Unlike 4G signals, 5G signals are transferred over short distances through a plethora of small, low-power base stations that can be located on light poles or rooftops. This is the primary difference between 5G and 4G signals. Since the new 5G mobile generation uses the low-frequency spectrum to transmit signals, the mobile network structure was built via radio operators. With 5G, signal transmission is virtually unaffected by the weather, building obstructions, or distance. The previous generation of wireless communications worked in a low-frequency spectrum band, so millimeter-waves [33] have to deal with interference and distance challenges that were not present during the last era. Wireless communications over 5G will be highly reliable, with ultra-low latency and very high throughput. For future and existing applications, 5G is expected to support a large number of wireless connections. Proximity service (ProSe) [34], a critical feature in 5G, allows for awareness of nearby devices and remote services and is provided through D2D communications. It also provides data management services for cloud computing, software-defined networks (SDN). The 3GPP has built a backward and forward compatible 5G standard using current LTE technologies [35] and will soon reveal new air interfaces for 5G technology. In [36], the authors discuss the new radio access technologies for connected and autonomous vehicles, such as visible light communication (VLC), millimeter wave (mmWave), C-V2X, and 5G, along with the challenges and opportunities. The new directions in research related to seamless connectivity, edge, fog, SDN, and security are also focused. An in-depth survey and comparison of vehicular MAC routing protocols was conducted in [37]. It addressed routing protocols with a cross-layer approach with MAC for VANETs. The MAC-aware routing solutions were classified as contention-free and contention-based. It provided a fully standardized, cross-layer communication model that is fully compliant with the existing vehicular service and application layers and messaging sets established by automotive and standards communities.

### 3. MAC Classification in VANETs Based on Contention Mechanism

Traditional classification of the MAC protocol in the literature [38,39] includes contention-based, contention-free, and hybrid protocols. Similarly, in this section, we studied some traditional MAC protocols in VANETs and classified them as contention-based, contention-free, and hybrid. This section ended by stating some of the recent advancement in vehicular communication, giving future directions.

**Contention-based protocols:** Commonly referred to as “listen before talk”. The nodes perceive the channel first; if it is free, it transmits. The system contends for channel access; the node that wins transmits through the channel. It is primarily used in a sparse network scenario so that the bandwidth can be utilized efficiently. These types of protocols cannot be used in real-time applications due to the time-bound and reduced throughput caused by collision. IEEE 802.11p and CSMA-based protocols are examples of this type of protocol.

**Contention-free protocols:** The nodes need not compete to gain channel access, and the time period of transmission in the channel is pre-allocated. Collision is avoided during data transmission in the channel. The transmission frames are divided into slots, and the nodes are synchronized for channel access. Although this type of protocol provides quality of service (QoS), it suffers from improper bandwidth utilization due to explicit time slot allocation. The main challenge is the proper allocation of channel resources among nodes. TDMA, FDMA, and SDMA are examples of this type of protocol.

**Hybrid protocols:** It merges the advantages of the contention-free and contention-based protocols to achieve QoS and to enhance network performance. It is primarily applicable for safety applications as there are minor delays and improved throughput. CSMA with TDMA [40], Token-Ring passing, and clustering-based protocols are some examples of this type of protocol.

ALOHA is one of the introductory MAC protocols [41] for radio packet networks. In ALOHA, there is throughput reduction for which S-ALOHA slotted ALOHA [42] was proposed, which splits the medium into various time slots and strives to transmit at the beginning of the time slot.

Multiple access with collision avoidance (MACA) [43] was proposed to overcome the hidden terminal problem using the handshaking method via RTS/CTS communication. The MACA for wireless (MACAW) [44] adds functionality to MACA to make it more robust when it comes to detecting collisions in WLAN data transmission. It requires nodes to send acknowledgements after every successful frame transmission. Similarly, BTMA [45] was proposed to overcome the hidden node problem by splitting the channel into data channels and a control channel. MC-MAC [46] uses two codes, one for control packet transmission and the other for data packet transmission.

ADHOC-MAC [47] uses a slotted frame and dynamic TDMA mechanism; R-ALOHA [48] was designed for dynamic TDMA in a distributed way. The reliable R-Aloha (RR-Aloha) [49] architecture was designed similar to ADHOC-MAC, which uses distributed MAC and UTRA-DD (UMTS terrestrial radio access time division duplex) for physical channel access with the single-hop broadcast. Directional antenna-based MAC [50] is yet another development in past years that uses the GPS of the terminals and restricts transmissions to a geographical area.

WAVE (802.11p) [51] mechanisms have no predetermined schedule, and channel access is random for vehicles, resulting in transmission collisions in a dense network scenario. 802.11p uses CSMA/CA; safety-critical applications cannot be guaranteed QoS due to its contention-based nature. Contention-based MAC protocols are designed to increase scalability under heavy loads by considering parameters such as the physical carrier sense threshold, contention window, and the transmission power control. WAVE employs the use of GPS to synchronize DSRC radios installed on all vehicles. Sync intervals (SIs) typically consist of CCH intervals (CCHIs) and SCH intervals (SCHIs), separated by guard intervals. Adaptive collision-free MAC (ACFM) [52] is a TDMA contention-based protocol with dynamic slot allocation. In ACFM protocols, unused slots are avoided during sparse traffic, and additional spaces are allocated in dense traffic.

The VeMAC [53] protocol supports multi-hop broadcasts and one-hop broadcasts on the control channel while reducing collisions caused by node mobility on the access channel. Vehicles moving the opposite way are assigned disjoint time slots and roadside units to reduce collisions during merging. Two transceivers are used per node, tuned to the control and service channels, and synced by the GPS signal. The VeSOMAC protocol [54] is a self-organizing, DSRC-based MAC protocol used in multimedia applications. The TDMA

slot information is exchanged in-band during distributed MAC scheduling. In highway scenarios, this enables fast reconfiguration of TDMA slots without depending on roadside infrastructure. VeSOMAC operates both synchronously and asynchronously. Through cooperative TDMA MACs [55], non-safety applications can achieve greater throughput. This protocol overcomes problems associated with poor channel conditions, which lead to transmission failures. If a packet fails to be transmitted, the neighboring nodes call helper nodes to relay the packet in a time slot. It is important to note that the helper nodes use the available time slots, which may lead to access collisions between the vehicles and the helper nodes. An enhancement of ADHOC MAC, the adaptive real-time distributed MAC (A-ADHOC) [56] protocol, intends for real-time application in large-scale wireless vehicle networks that provide adaptive frame lengths. Both the channel resource utilization and response time of A-ADHOC were better than those of ADHOC MAC, and A-ADHOC avoided network failure regardless of traffic density.

Some of the hybrid protocols (TDMA + CSMA/CA) include HER-MAC (hybrid efficient and reliable MAC), TDMA/CSMA MAC (HTC MAC), SOFT-MAC (space orthogonal frequency–time MAC), and DMMAC (dedicated multi-channel MAC with adaptive broadcasting). The SDMA, OFDMA, TDMA, and CSMA techniques were incorporated into SOFT-MAC [57], with GPS to locate sub-carriers shared between vehicles belonging to that particular cell. The HER MAC [58] is a multi-channel MAC with adaptive broadcasting. Every vehicle in the CCH transmits safety/alert messages using a half-duplex transceiver without colliding with another vehicle. HTC MAC [59] alleviates the collision and enhances the throughput of HER MAC by broadcasting the announcement packet (ANC) and reservation period (RP). DMMAC [60] is also a dedicated multi-channel MAC with access time split into intervals (CCHI and SCHI), again split as contention-based reservation period (CRP) and adaptive broadcast frame (ABF).

This discussion presents some of the traditional MAC protocols classified based on contention mechanism as contention-based, contention-free, and hybrid presented in Figure 2. Furthermore, the next section provides a brief classification of contemporary MAC protocols in vehicular communication.

A summary of different MAC protocols is shown in Table 2.

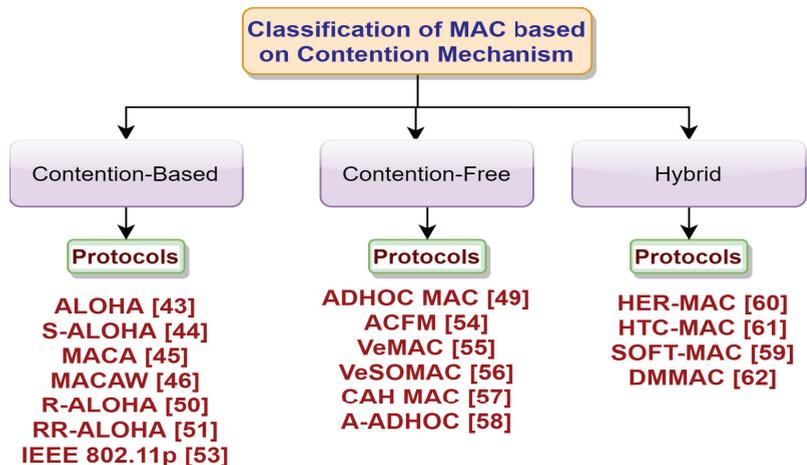


Figure 2. Classification of MAC protocols based on the contention mechanism.

**Table 2.** Summary of traditional MAC protocols.

Protocols	Mechanism	Objective	Disadvantages
ALOHA [41]	TDMA based	Collision avoidance	Throughput reduction
S-AIOHA [42]	TDMA and CSMA sense the channel before transmission, and CSMA/CD used for collision detection	Collision avoidance and improved throughput	Unable to avoid collision in high-density scenarios and cannot handle hidden and exposed terminal problems
MACA [43], MACAW [44]	Handshake method by RTS/CTS and ACK	Overcomes hidden terminal problem	Performance degradation in dense scenarios
IEEE 802.11p [51]	DCF, CSMA/CA, WAVE, and DSRC-based	Reliable, adapted to frequent changing VANET environments, and alleviates collision	Applicable to minimal distances, and no focus on hidden and exposed terminal problems
ADHOC-MAC [47]	Dynamic TDMA based-distributed protocol	Contention-free access, and multi-hop and one-hop broadcast with QoS	Transmission overhead, inefficient real-time traffic control, and transmission delay
ACFM [52]	TDMA, and dynamic slot assignment	Alleviate collision	Lacks real-time control, and QoS degrades in dynamic scenarios
VeMAC [53]	TDMA based, and one-hop and multi-hop broadcasts	Handles hidden/exposed terminals	Delays due to broadcast storms
VeSOMAC [54]	TDMA-based distributed scheduling	Improved throughput in highway scenarios	Channel utilization is inefficient in high densities
CAH MAC [55]	Distributed TDMA	Enhanced throughput (non-safety application)	Access collision problem
A-ADHOC [56]	Adaptive frame length	Proper channel utilization and real-time application	Less throughput due to overhead
HER MAC [58]	Multi-channel access and adaptive broadcasting	Channel split and three-way handshake	Collision problems and delay
HTC MAC [59]	Hybrid TDMA/CSMA multi-channel	Overcome collision	Hidden/exposed terminal problems and communication overhead
SOFT MAC [57]	SDMA, OFDMA, TDMA, and CSMA mechanisms	Proper channel and resource allocation	Highly complex with overheads
DM MAC [60]	Dedicated multi-channel, co-ordinated adaptive broadcast	Proper broadcasting and throughput maximization	Performance degradation in changing topologies due to random slot assignment

In recent years, MAC protocols with dynamic interval schemes [61] have been proposed for the optimization of channel control intervals. A MAC protocol for dynamic adaptation due to changing VANETs topology is a challenge as the static service channel and the control channel cannot adapt to the versatile VANETs environment. Protocols designed with a dynamic interval scheme provide maximized performance with less end-to-end delay and minimized collision.

Clustering is a prime concept in VANETs for efficient group communication. Some VANETs integrates machine learning and fuzzy logic algorithms to stabilize and make clusters more efficient. The clustering algorithms are classified into three broad categories: intelligence-based strategies (machine learning algorithms, fuzzy logic algorithms, and hybrid algorithms), mobility-based strategies (NEMO algorithm and mobility algorithm), and multi-hop-based strategies (2-hop algorithms and 2+-hop algorithms) [62]. Some other clustering-based algorithms are proposed in [63,64].

The SDN-based network is one of the technologies used recently in most fields for reliable and efficient communication; it splits the control plane from the data plane by providing centralized access to network resources. Reference [65] presents a concise layer

structure of VANETs along with an SDN controller. The application of SDN-VANETs on various parameters in IoT and wireless communication is presented. The open challenges and research directions faced by the latest work, including VANET integration with SDN; the recent and emerging technologies; as well as the use cases are demonstrated in [66].

With the new standards described in Section 2, it is possible to transition from “legacy” IEEE 802.11p systems smoothly. By implementing the same frequency channel worldwide, IEEE 802.11bd takes advantage of existing deployments and infrastructure without interfering with current ITS applications. IEEE 802.11bd achieves interoperability through a compatible waveform structure. In addition, it employs a well-known channel access mechanism, “listen-before-talk” (carrier sensing) as asynchronous and non-persistent V2X network communication is substantially more flexible in size and transmission rate than conventional V2X networks [67]. The IEEE 802.11bd Next Generation V2X Study Group was formed in March 2018 to minimize the performance gap between DSRC and C-V2X, to support additional modes of operation, and to increase throughput [68].

#### 4. Classification of MAC Protocols Based on Channel

This section presents the classification of MAC protocols in VANETs; oriented as distributed, centralized, cooperative, cluster-based, virtual token, and random selection based and further sub-divided as multi-channel and single-channel. The classified MAC protocols based on channel access mechanisms are depicted in Figure 3 and discussed.

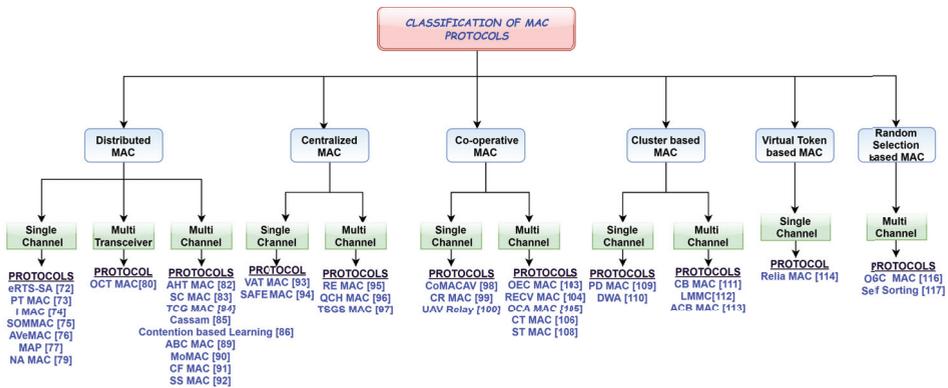


Figure 3. Classification of MAC protocols.

##### 4.1. Distributed Single-Channel MAC Protocols

There is no centralized assistant for channel access. In a single-channel MAC, the access channel is not divided, so there can be interference chances. These are basically designed to resolve challenges related to packet allocation and resource allocation based on TDMA, CDMA, and SDMA. The main focus is on TDMA- and SDMA-based time-slot allocations and contention-based parameters for improving network throughput from that of IEEE 802.11p. To adapt to VANETs’ dynamic change in topology, distributed MAC protocols are designed to minimize communication overhead.

###### 4.1.1. eRTS-SA

The request transmission split-slotted ALOHA-based MAC protocol provides fair channel access with high throughput and interference avoidance using a GPS-embedded micro-cell unit.

Every vehicle reports their location information to micro base stations (mBS) by GPS, and MEC is used in the contention access phase to provide cloud capabilities. The road between mBS is divided into the segment and allocated with segment number to detect location information to reduce overheads. The vehicles are arranged for time slot allocation

in increasing order of their segment number. To minimize channel interference, the same segment number is allocated to vehicles geographically apart. The methodology includes dividing the frame into three phases: contention access phase (CAP), broadcast feedback phase (BFP), and contention-free phase (CFP). In CAP, the vehicles send a request signal to RSUs. After the signal is received by RSU, successive interference cancellation (SIC) is implemented; then, the vehicle occupies the time slots to enable transmission without interference. To prevent interference, the BFP is divided into two divisions assigned to adjacent mBSs. The mBS are equipped with the MEC server to serve vehicles within their micro-cells, and vehicles move in a bi-directional highway with no “on-and-off-ramp” [69].

The physical interference model based on signal-to-interference-plus-noise ratio (SINR) is calculated to keep track of successful transmission. Vehicles and mBSs are time-synchronized; the time-slot allocation mechanism avoids the hidden terminal problem. This protocol is better in throughput (4.8 percent lower than the theoretical maximum throughput) and transmission efficiency. The protocol does not prioritize safety-based applications or non-safety-based applications, energy efficiency is not analyzed, and the scalability issue is not addressed.

#### 4.1.2. PTMAC

This protocol is used for collision prediction and collision reduction effectively in two-way traffic as well as in four-way intersections. The protocol revolves around three steps: potential collision detection, potential collision prediction, and potential collision elimination. Potential collision is detected based on slot information; the prediction of collisions in the future is based on traffic and vehicle information conditions. The potential collision can be eliminated by rescheduling the slots (mostly tackling encounter collision). Every node broadcasts its information, including location, slot reserved, speed, and direction. The frame information (FI) contains information about its one-hop neighbor nodes and itself. The new joining node has to listen to the channel before contending for slot allocation. Potential collision detection in same and opposite directions is depicted in Figure 4.

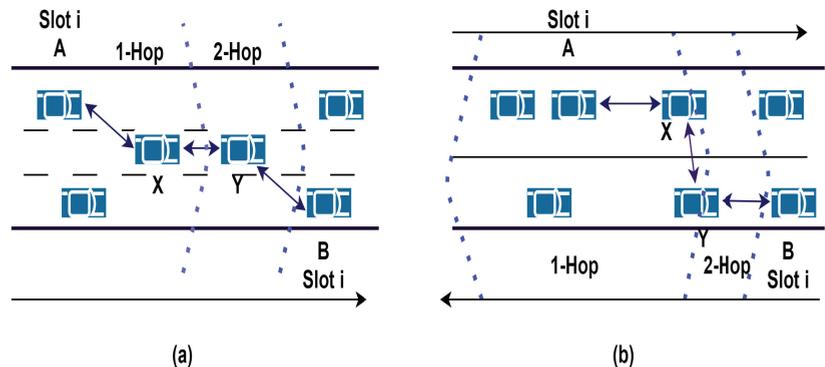


Figure 4. Potential collision detection: (a) same direction; (b) opposite direction [70].

In potential collision detection, every node checks whether the same slot is occupied by any of its one-hop or two-hop neighbors. The intermediate nodes contain slot reservation information and detects potential collision between the two one-hop neighbors. After detecting potential collisions, the nodes that collide are allocated different slots and broadcast messages to their one-hop neighbor.

There is no slot partitioning; thus, unbalanced traffic density has no impact on the performance. PTMAC effectively handles access collision as well as contention collision. However, network overhead and power consumption need to be studied more.

#### 4.1.3. I-MAC

Improved MAC (I-MAC) avoids collision in high-density VANET scenarios via proper channel access. It reduces the loss of messages due to efficient channel utilization. It employs a dynamic TDMA-based mechanism with CSMA. An initial broadcast table (IBT) containing the vehicle's MAC address is created before sending data. Due to the possibility of more than one vehicle sending to IBT simultaneously, causing a collision, the nodes sense the channel before sending to IBT. A collision may even occur if nodes start transmitting as soon as it senses an idle channel; therefore, I-MAC uses dynamic inter-frame spacing (DIFS), which means nodes wait for a random time before transmitting through an idle channel to provide fair channel access.

The protocol avoids merging collision and access collision, thus improving the channel's performance and the reliability of data transmission. There is much communication overheads, and the hidden node problem analysis needs further investigation.

#### 4.1.4. SOMMAC

A self-organizing multi-channel protocol was proposed for rapid handover in the network without disconnecting and maintaining the performance level of the network. It aimed to alleviate access to collisions, to improve the packet delivery rate, and to minimize delay.

It uses DSRC for multi-channel transmission in two-way traffic. The vehicles listen to channels to verify whether there is a message from the scheduling channel (SCCH). If no message is found, the vehicle waits and sends a joining request from the contention-based sub channel (CBCH) by CSMA. The RSU that controls the network frequently sends RSU heart-beat packets (RSUHB) from SCCH to let other vehicles know that the vehicles receiving data packets are near the RSU, thus avoiding collision. Upon accepting a channel request packet (CRP) from CBCH, the RSU assigns a service channel based on the direction of the vehicle. It assigns time slots to vehicles based on vehicle direction. RSU consists of a vehicle channel information table, which holds the vehicle's information. A configuration packet is prepared consisting of channel information and sent to the destination.

As soon as a vehicle receives a configuration packet (CP) belonging to itself, it fixes the channel assigned for transmission and releases all other channel and time slots. The vehicle is then added to the network and sends vehicle heart-beat (VHB) packets from the heart-beat channel (HBCH) based on its direction. Upon receiving VHB, the RSU checks the vehicle information table; if the vehicle is present, then it checks the time period in which the vehicle should be present in the network. After the vehicle drops this time, it is removed from the network so that other vehicles can assign themselves to the released time slot. The dynamic channel assignment method is used to improve efficiency and fairness in accessing the channel based on TDMA, with a better packet delivery ratio and minimized latency. The main challenge includes power consumption and handling multi-directional scenarios with minimized communication overhead.

#### 4.1.5. AVeMAC

Adaptive vehicular MAC is proposed to adaptively vary the channel condition and to handle unbalanced traffic in the opposite direction in a two-way intersection road scenario. It also eliminates merging collision and access collision and improves channel utilization. AVeMAC is an enhanced VeMAC [53], which is a TDMA-based protocol supporting one control channel and several service channels.

The frames in the control channel are partitioned into disjoint sets of time slots analogous to moving vehicles in the left or right directions [71]. The partitioning of the channel is not fixed as in VeMAC; rather, it varies adaptively to the traffic condition. Here, vehicles cannot use the time slots occupied by vehicles one-hop or two-hop away but can use the slots occupied by at least three-hop neighbors. Before adapting to timeslot partitioning, vehicles need to listen to the channel for  $N$  successive time slots to obtain information about a set of one-hop and two-hop neighbors. Then, the set of timeslots available for allocation can be determined so that the vehicle can randomly select a time

slot to be allocated for further transmission. It then checks whether the reservation of that slot is successful by listening to the channel for the next  $N-1$  slots. Suppose that the reservation is found to be unsuccessful. In that case, it is estimated that some other vehicle in a two-hop neighbor attempts to reserve that slot, so the vehicle tries to reserve a new slot, thus avoiding access collision. The vehicle continues to access the same time slot until a merging collision occurs detected based on frame information (FI) messages.

The protocol improves channel utilization. However, it does not consider the hidden terminal problem. QoS matrices such as reliability, packet delivery ratio, and network throughput are also not analyzed.

#### 4.1.6. MAP

Medium access for PLNC is a contention-free MAC protocol used to provide quick and reliable data transmission in VANETs. It addresses two fundamental problems: collision due to hidden terminals in CSMA/CA-based networks and handling excess control messages in dense TDMA-based network scenarios. It is a decentralized location-based scheme, where the network is divided into sub-zones of equal length. Vehicles incorporate CSMA/CA schemes to access the channel; the successful node transmits and is called the intra-zone relay node [72]. The transmission occurs in two phases: the access phase, when nodes transmit packets to relay nodes, and the broadcast phase, where the relay forwards the network-coded packet. MAP handles the hidden terminal problem by priority-based indexing [73]. To prevent dissemination delays, a session order prioritization scheme is implemented. It provides reliable and faster transmission.

There is a possibility of merging collision as this protocol only handles access collision; delay, Packet Delivery Ratio, and network throughput are not analyzed. The network model presented is based only on a 1D network.

#### 4.1.7. NA MAC

A novel neighbor association-based MAC is designed for reliable broadcast. The protocol is based on TDMA with the CSMA approach for disseminating basic safety messages (BSM), providing a short time for each slot. Duplicate slots allocated are avoided by the implementation of communication via a three-way handshake. The frame structure of NA MAC for collision avoidance is depicted in Figure 5. TDMA-based broadcast is provided by V2V and V2I communication. The protocol alleviated collision and hidden-terminal problems, providing reliable communication with less overhead and a good packet reception ratio. It also minimizes the latency of safety message transmission.

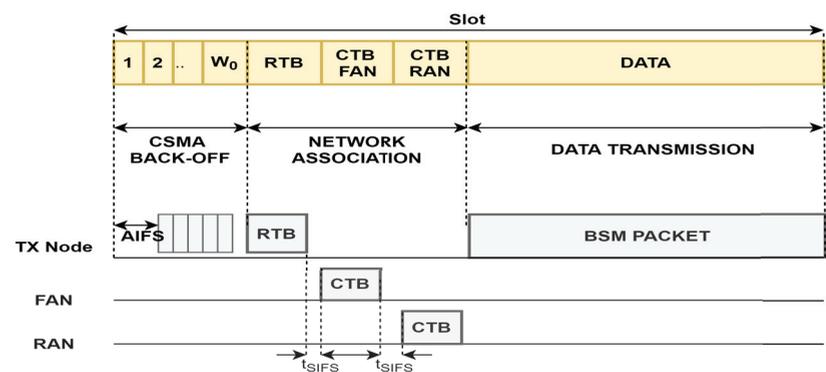


Figure 5. Frame structure of NA MAC for collision avoidance [74].

This protocol's shortcoming is that it does not emphasize non-safety message transmission; single-channel access leads to improper resource utilization when considering both safety and non-safety messages. A broadcast storm problem causes performance degradation.



time and, thus, improves throughput and minimizes delay. The demerit of this protocol is the broadcast overhead due to DTR/DTS in dense and large networks with packet loss.

#### 4.3.2. SCMAC

The slotted contention-based MAC protocol addresses two types of collision. First, one occurs due to nodes not being present in close proximity, and the other is due to hidden terminals. The first is overcome by the back-off scheme of IEEE 802.11p, and the latter is overcome by TDMA. Here, the time slot is divided into two periods, i.e., TP (transmission period) and RP (reception period). During the TP, a node broadcast packet is needed to transmit, whereas the channel is reserved in the RP. If there is a reservation failure, the node does not transmit the packet in the TP but dedicates another time slot for further transmission. In SCMAC, black burst-based slot reservation is used to jam signals, thus allowing only one node to broadcast the message at a particular TP. The hidden terminal problem is overcome by utilizing spatio-temporal co-ordination (STC). The road is divided into four segments with fixed lengths, and different slots are assigned to the segments.

SCMAC outperforms other back-off schemes in terms of packet delivery ratio. The use of STC is suitable for highway scenarios but not for rapid topological changes in VANET, as it is difficult to divide the road into segments as in STC. Thus, there is some transmission delay and performance degradation even if interference is avoided.

#### 4.3.3. TCG MAC

The TDMA-based MAC with the collision alleviation protocol combines TDMA plus CSMA-based MAC for collision mitigation. The methodology of game theory is implemented when more nodes acquire the same slot as in the CSMA period for efficient slot allocation. This reduces access collision. Vehicles are equipped with GPS to know the position and direction. Synchronization among vehicles is performed by the 1PPS signal administered by a GPS receiver.

The nodes are divided into the direct neighbor set (DNS) belonging to a one-hop neighbor of the transmitting node and the indirect neighbor set (INS), those not a direct neighbor but within the communication range and other nodes that are not in the transmission range. Upon broadcasting the packet, the vehicle ID and slot reservation information of each node belonging to DNS are attached to the header; this enables the nodes to know the slot allocation information of its two-hop neighbors. The nodes update their neighbor set based on information from the DNS. Only collision nodes send reservation messages in CP when they hold a TP slot, and each node must acquire exactly a single-time slot in TP to broadcast messages. The transmission ranges of R and 2R are broadcast by nodes in TP and CP, respectively, as depicted in Figure 7. For all relative neighbor set (RNS) nodes to receive the slot reservation message transmitted by CP, it must have a longer transmission range. The strategy of game theory alleviates collision; this protocol shows high throughput with proper allocation of slots in different channel conditions. There is high power consumption due to the use of transmitters and receivers. There is high network overhead due to the game theory approach.

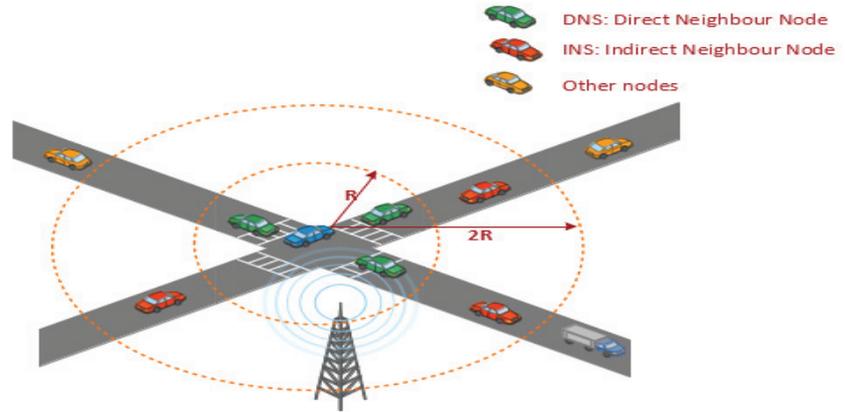


Figure 7. Mechanism of TCG MAC [78].

#### 4.3.4. CaSSaM

The author in [79] proposed a context-oriented information-based system. It is a dissemination-based protocol that works in a decentralized environment. This protocol considers the channel busy ratio (CBR), collision, number of neighbors, speed, and inter-vehicular distance. The CaSSaM system decides which parameter is adjusted for performance enhancement. The slotted 1-persistence protocol is used for the dissemination mechanism. When a node receives a packet for the first time, a time slot is allocated to the node and the node is then re-broadcasted with a probability of 1; for re-broadcast, priority is given to the farther node.

It is mostly used for guaranteed safety applications with reduced delay and enhanced throughput. The power consumption of nodes is not analyzed; the protocol has a decentralized approach, and performance can be improved by implementing a centralized approach or by a combination of both.

#### 4.3.5. Contention-Based Learning MAC Protocol

The primary objective is to increase the network scalability, to reduce the bandwidth usage, and to minimize packet collision in a dense topology. The machine learning-based reinforcement learning (RL) [80] technique is used. A self-based Q-learning technique, suitably used based on RL, improves performance by controlling contention in the network. The distributed coordination function of the IEEE 802.11p standard is used with the CSMA/CA mechanism for both unicast and broadcast. The back-off mechanism is used to check whether more than one node accesses a channel at the same time. Q-learning [81] and a Markov-decision process are used to avoid packet collision using an optimized CW based on binary feedback.

The protocol minimizes latency, improves the packet delivery ratio and throughput of the network. There is performance degradation due to a high-mobility, dynamic environment. The fairness in transmission and latency still need be improved.

#### 4.3.6. ABC MAC

The adaptive beacon control protocol is proposed to avoid collision at the rear ends due to congestion. The authors proposed an adaptive beacon rate scheme based on rear end collision, considering the kinematics status of adjacent vehicles and considering a danger co-efficient  $\rho$ . Based on the bandwidth requirement and channel capacity, a distributed beacon rate adaptive (DBRA) problem is formulated. A vehicle bearing high  $\rho$  estimates is assigned high beacon rate to avoid collision. During congestion, the vehicle adopts a greedy algorithm to solve the DBRA problem and TDMA-based broadcasting is conducted for

neighboring vehicles. The protocol works in three basic steps: (i) detection of congestion, (ii) adaptation to distributed beacon rates, and (iii) broadcasting the adapting result to other vehicles.

The protocol guaranteed high performance in dynamic traffic scenario alleviates collision and provides an efficient transmission ratio. QoS in terms of throughput can still be improved. However, it does not emphasis non-safety messages. The off-road collision, real-time traffic management is not analyzed.

#### 4.3.7. MoMAC

The mobility-aware MAC protocol was proposed to achieve collision-free transmission and to enhance the message delivery and reception rates of safety applications. It handles the hidden terminal problem. TDMA-based slot partitioning is used; vehicles use GPS to synchronize and use the same slot until collision is detected. Different time slots are selected by vehicles within the same OHS. To resolve the hidden terminal problem, vehicles in the same THS should choose different communication time-slots. During a THS, the hidden terminal problem can occur when two vehicles, located in the two OHSs, cannot hear each other and decide to send messages simultaneously. For instance, in Figure 8, vehicle A wishes to transmit a message to vehicle B, and vehicle C wishes to transmit a message to vehicle D at the same time. Vehicle C does not realize that vehicle A has already started transmitting because it is not within its communication range, which results in a collision with vehicle B. Since there is no RTS/CTS mechanism, each vehicle should collect and broadcast information about all other vehicles occupying time slots from one-hop neighbors to the other THS, so there is no hidden terminal problem. The protocol is implemented in a multi-lane road segment where the segment is divided into time slots. The collision detection scheme and distributed slot access are employed to eliminate hidden terminal problem [82].

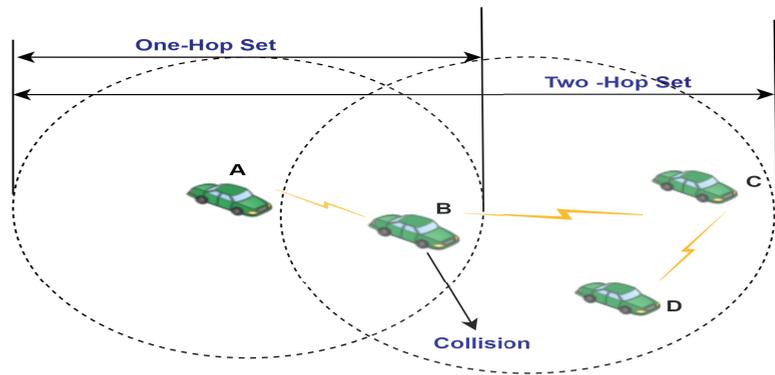


Figure 8. Illustration of hidden terminals [82].

The protocol aims to minimize delays in access to and delivery of packets. The communication is decentralized, not RSU centric. Therefore, current traffic detection is not easy and cannot handle dynamic traffic scenarios.

#### 4.3.8. CF MAC

The main objective of the collision-free MAC protocol is to make collision-free transmissions, improving the performance and reliability of channel access. Here, an initial broadcast table (IBT) aligns MAC addresses of vehicles in ascending order for setting priorities for transmission. The IBT is updated and broadcast periodically to share the status of the channel. A receiver checks the IBT for the initiator from the first slot; if it wants to send it, it sends a WTS to the initiator. Upon reception, the WTS initiator adds 1 to the IBT. The priority in sending is highest in the vehicle that is at the first slot of BT,

so the other vehicles synchronize themselves according to the information of a vehicle's priority received from BT [83].

This protocol eliminates mergers and access collisions for better performance and reliable transmission. There is a delay in transmission caused during contention avoidance, as vehicles have to wait for a random amount of time before sending. Performance degrades in dense scenarios due to transmission delays.

#### 4.3.9. SS MAC

Slot-sharing MAC was proposed to provide a scalable, reliable, efficient protocol with less delays for safety message broadcasting. Multiple vehicles broadcast alternately on the same time slot using inconsistent coordinations. A circular recording queue is implemented to record the time slot status of the periodic broadcast of safety messages. A distributed time slot sharing (DTSS) mechanism is designed to check on the periodical broadcasting of messages and to share the time slots for reliable transmission. To improve the channel utilization based on heuristic packaging strategy, a random index fit-first (RIFF) scheme is proposed [84]. This assists the vehicles in selecting a suitable time slot for sharing. It employs a traditional TDMA-based mechanism with a slot sharing scheme.

The major demerit is that the protocol cannot handle dynamic resource environments, and safety and non-safety applications. There is overhead for maintaining the table and broadcasting.

#### 4.4. Centralized Single-Channel MAC Protocols

It is either RSU based or cluster head (CH) based. RSUs allocate time slots to the nodes for channel access. It is a better mechanism for providing collision-free access as scheduling of time slots is not as complicated. The periodical broadcasting mechanism employed by RSUs minimizes collision by communicating the time slot reservation messages to nearby vehicles.

The coordination is wholly dependent on either the RSU or the CH. The data are transmitted by a time-synchronization mechanism to avoid collision, but there is network overload due to the centralized control.

##### 4.4.1. VAT MAC

According to VAT-MAC (novel adaptive TDMA-based MAC) protocols, RSU is used to provide efficient access management for more efficient network performance [85].

In the VAT-MAC time management period (TMP), the RSU broadcasts a time management frame (TMF), noting the length of the free transmission period (FTP) and contention period (CP) depicted in Figure 9. The FTP protocol is used for packet transmissions of allocated time-slots, and unidentified vehicles can compete for idle slots in the CP. The RSU can identify the assigned slot in the upcoming FTP if the vehicle successfully accesses the CP without colliding. The RSU further calculates the average vehicle density based on the collision probability in order to anticipate the number of newly entering vehicles. By doing so, it is able to predict the collision probability. By adjusting the frame length accordingly, VAT-MAC is capable of improving network scalability and of ensuring the efficient use of time slots. The mathematical analysis and simulation experiments indicate that VAT-MAC can significantly enhance system scalability and throughput. The performance can be enhanced with incorporation of CSMA/CA along with TDMA mechanism.

##### 4.4.2. SAFE MAC

The speed-aware fairness-enabled MAC [86] is a RSU-based centralized approach utilizing the CSMA/CA mechanism with dynamic adjustment of CW, a back-off, and a re-transmission limit. Based on the mobility metrics, including speed, location, and direction, the system computes the time spent by vehicles in the service area. Moreover, the vehicles are divided into three groups according to their duration of stay. Each batch has its own MAC parameters. These parameter values are then dynamically changed to ensure that the vehicles with higher speed receives a certain minimum number of messages, which

can guarantee fair channel access for V2I. The main limitation of this protocol is it does not consider fairness issues in channel access for V2V and V2D communications.

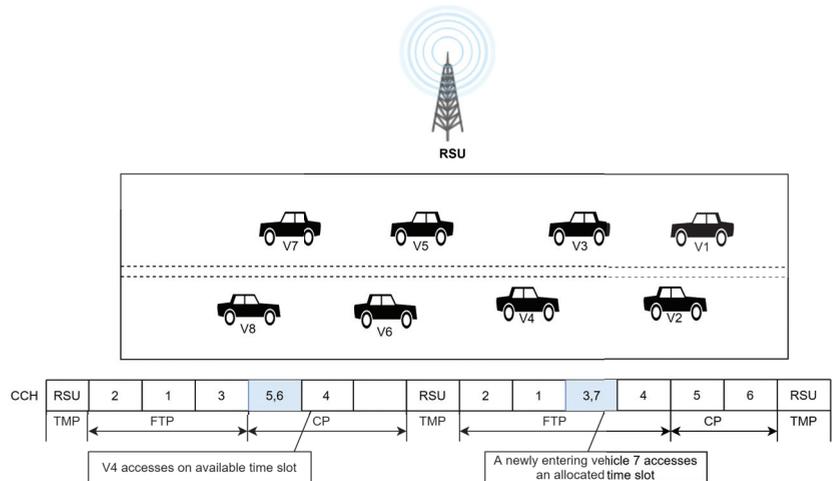


Figure 9. Time frame structure of VAT MAC [85].

#### 4.5. Centralized Multi-Channel MAC Protocols

Control is centralized by a central coordinator, but the channel is partitioned to deal with safety and non-safety messages by prioritizing it to minimize latency for safety messages.

##### 4.5.1. ReMAC

The main objective of the reliable MAC protocol is to connect vehicles with a multi-channel hybrid medium of access. It reduces collision and hidden terminal problem and deals with high mobility by randomly minimizing transmission delay. Here, the modeled system consists of RSUs placed at regular intervals in a two-way highway with vehicles moving at high speed. For determining the next RSU, the vehicle’s direction is considered, and the channel is assigned before it enters the range of next RSU. FDMA-based access is used, dividing the CCHs and SCHs into a sub-frequency channel of same bandwidth. This mechanism avoids collision and provides reliable and efficient communication. Each sub-frequency channel are divided into time slots by TDMA. Figure 10 demonstrates the network model or structure of ReMAC.

The separation of a guard band is chosen to be wide to minimize the channel interference. A joining network channel (JNCH) for new vehicles is used when they initially enter the network; then, the request is sent to RSU from JNCH by CSMA. The vehicle at the beginning adjusts to communicate with control channel and data channel and then sends a channel request packet to RSU by CSMA from JNCH. The RSU upon receiving VSIP [87] allocates a data channel to the vehicle and time slots from convenient channel, taking into account the moving direction of vehicles. The updates are provided in the VCAT table. When RSU receives a package present in the private network within its range, it searches for the vehicle’s information in the VCAT table. The table is updated on a regular basis and channel is allocated per the requirement. The calculation of network drop time (DT) is carried out to check whether the vehicle is dropped out of the network or still present, and the vehicles not sending heartbeats within 0.5 s are disconnected. The channel and slot allocation algorithm is based on FDMA by selecting a backward frequency band and a forward frequency band.

The protocol provides better performance in channel access rate, collision avoidance, high throughput, and minimized delay in dense city scenarios. It also improved the scalability of the network and adapted to frequent diverse topological changes. There is

communication overhead (at RSUs and OBUs) along with high energy consumption that need to be optimized.

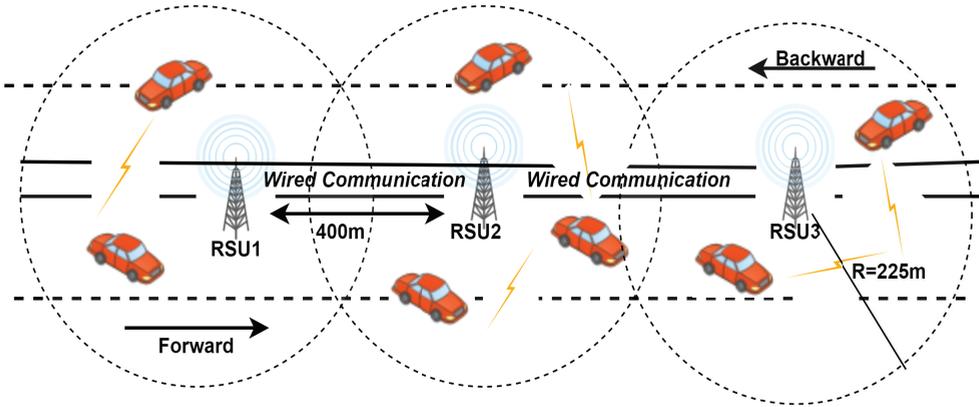


Figure 10. Network model of ReMAC [87].

4.5.2. QCH MAC

The QoS-aware centralized hybrid protocol avoids collision and provides QoS in terms of transmission delays, throughput of the network, and packet delivery. The protocol combines an extended version of EDCA and TDMA. The transmission of safety-based messages is prioritized based on slot scheduling mechanism by RSUs. The access time is divided into two periods: the transmission and reservation periods. The transmission period uses time slots for scheduling, and the reservation period is used only by new vehicles to reserve the slot as soon as it enters the traffic scenario.

Safety and non-safety messages are treated differently; safety messages are considered higher priority, denoted as CL1, than non-safety messages, denoted as CL2 [88]. The vehicles that enter the transmission initially sense the channel and wait for the medium to become free for transmitting. If the channel is busy, the back-off mechanism is used to prevent collision. Second, after reservation of the slot, the vehicles enter the transmission period and are ready to transmit. The mechanism is depicted in Figure 11.

The major demerit is that the protocol does not deal with hidden terminal problem and throughput can still be improved using optimized scheduling.

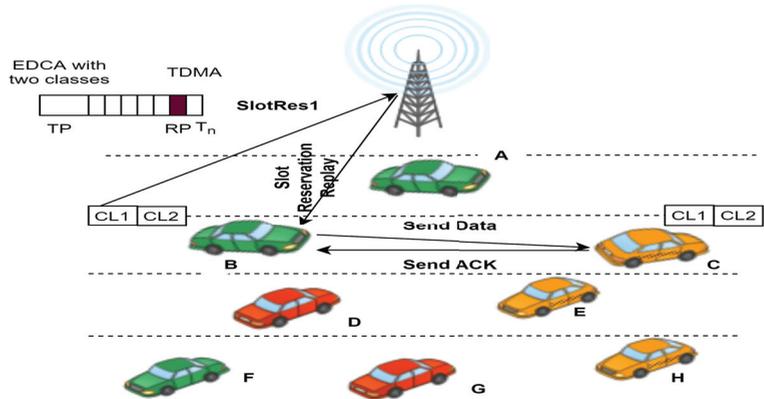


Figure 11. Access mechanism of QCH MAC [88].

### 4.5.3. TSGS MAC

A Transmission Scheduling Greedy Search (TSGS) algorithm provides a contention-based scheduling mechanism with less time complexity. The algorithm is based on setting a time for each transmission with CSMA/CA. By minimizing the overlap of transmission, the number of collisions is reduced, thereby minimizing the probability of activation of the back-off timer.

The implementation of TSGS provides an optimal time slot for connection, thereby providing better packet delivery ratio. The major demerit is the protocol designed is RSU centric, so there is communication overhead on RSU. The protocol does not address hidden terminal problem and scalability when the number of vehicles increases in the network.

### 4.6. Cooperative Single-Channel MAC Protocols

It basically deals with packet relay from the source to destination in an cooperative way. It improves the reliability of transmission and the throughput of channels.

#### 4.6.1. CoMACAV

Cooperative MAC was proposed for autonomous vehicles to provide high network throughput. There are three modes of data transmission on which the protocol works. These are direct transmission (DT), cooperative relaying (CR), and multi-hop relaying (MHR). The main objective of this protocol is to increase throughput by selecting the optimal value of relay (R) produced by the SNR concept. The DT (RTS/CTS) and CR modes of transmission are implemented when nodes are present in the same network. In contrast, the MHR mode is implemented when nodes are in different networks. New control packets such as RRTS, RH, RRR, RCR, and RACK are introduced and depicted in Figure 12.

All neighboring nodes through RTS/CTS know the SNR of both source and destination nodes. Thus, neighboring nodes that are free then broadcast optimal relays. The node with the highest SNR becomes the optimal relay node and sends RRR to both the source and destination. This increases the reliability of transmission. The analysis is performed through the Markov model. The demerit of this protocol is a delay in transmission due to collision.

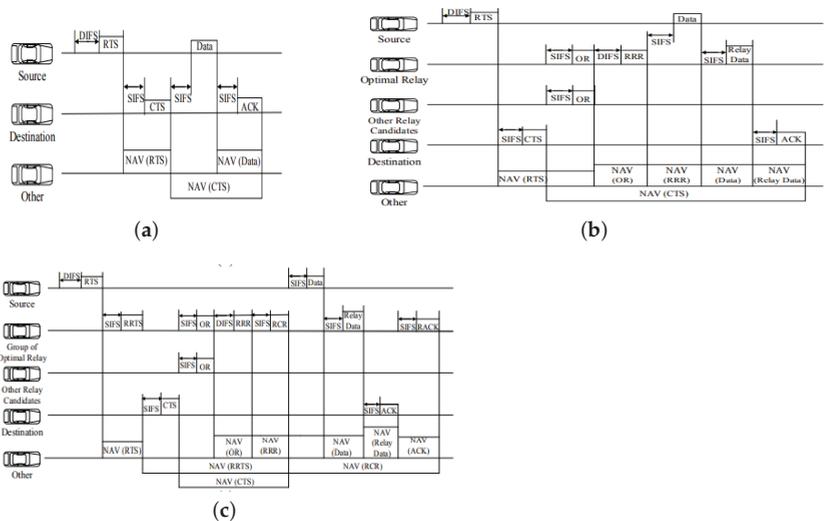


Figure 12. Mechanism of packet exchange: (a) DT mode, (b) CR mode, and (c) MHR mode [89].

#### 4.6.2. CRMAC

A cooperative-based protocol is proposed, where the three types of data transmission modes employed are direct transmission, cooperative relaying, and multi-hop relaying to achieve good throughput. Cooperative communication is performed via optimal relay selection. The RTS/CTS mechanism is used in a direct transmission (DT) mode. In the cooperative relay (CR) mode, all nodes know the sender and receiver SNR information (nodes with higher SNRs are chosen). The sender sends data to the optimal relay and then optimal relay sends it to the destination. If the sender does not receive any ACK after being sent by a short inter-frame space (SIFS) sender, the relay re-transmits the data to the destination. Nodes transmit with an interval of dynamic short-frame space (DIFS). Upon receiving a RACK, a sender knows that the transmission was successful. The mechanism of exchange for the packet is depicted in Figure 13.

It is basically a relay selection algorithm proposed to select the optimal relay based on the highest SNR among nodes. A performance analysis was performed based on the probability calculation, average transmission time, and Markov-Model used for analysis of the result. The protocol provides enhanced throughput and efficient communication. The demerit of this protocol is the lack of performance analysis on collision avoidance, power consumption, delay factors for real-time applications, and efficient use of channels.

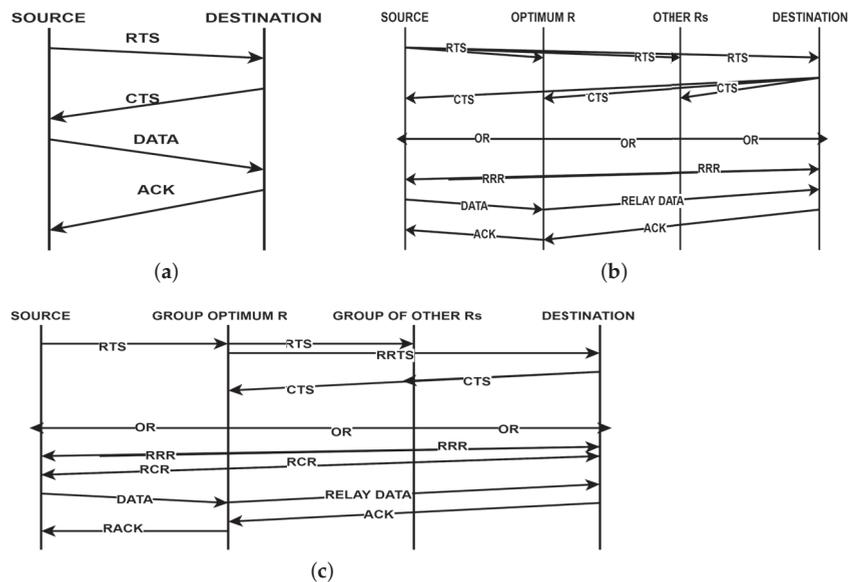


Figure 13. Mechanism of packet exchange: (a) DT mode, (b) CR mode, and (c) MHR mode [90].

#### 4.6.3. UAV Relay

The unmanned aerial vehicle-based MAC protocol was proposed for performance enhancement using a relay strategy. It improves the efficiency of communication hampered by interference or jammers at RSUs. UAVs use specific relay strategies to send messages from OBUs to RSUs. The Nash equilibria strategy was implemented, and UAV relay was optimized based on transmission cost and a UAV channel model. A hotbooting policy hill climbing (PHC) mechanism (a reinforcement learning technique) was used in the UAV relay strategy to resist jamming signals. The implementation of the network model was based on the calculation of bit error rate (BER) and SINR. Anti-jamming transmission stochastic game strategy (an extended work of [91]) was used depending on the quality of the channels and BER; this strategy minimizes the power consumption by making decision to choose the relay and whether to transmit. Dynamic anti-jamming game with

the hotbooting PHC-based relay strategy was used to formulate the interaction between UAVs and jammers [92], determined the jamming power, and selected the relay based on the state of the system.

The drawback of this protocol is communication and computation overhead. Highly mobile nodes and dense scenarios are not analyzed for this mechanism.

#### 4.7. Cooperative Multi-Channel MAC Protocols

In this type MAC protocol, the capabilities of multiple channels and cooperative communications are integrated for performance enhancement at the MAC layer.

##### 4.7.1. OEC MAC

A novel OFDMA-based efficient cooperative (OEC) MAC protocol was proposed by providing efficient sub-carrier channel assignments and access mechanisms. The concept of OFDMA was introduced to handle the delay and collision problem in high-density traffic, which could not be overcome by simple CSMA/CA. ACK, CRM, and CAM are used for safety messages (sm) and CWSA for non-safety messages (nsm). ACK is assigned for sm: if no acknowledgment is received, then there is no packet transmission; then, for successful delivery, the packets need to send with cooperative communication (broadcast CRM). This mechanism provides a reliable transmission broadcast service. A node with higher SINR, transmission rate, and channel condition sends back CAM, and the optimal relay is selected from nodes sending CAM. Once ORM selects the optimal relay, other nodes are held back from sending CAM. For the delivery of non-safety messages, WAVE communicates with periodic broadcast of WAVE service advertisement (WSA). If the relay node has better channel conditions and SINR, CWSA, which consists of the relay information, WSA information, and channel information, is broadcast. The optimal relay is chosen after receiving CWSA by broadcasting ORM. Lastly, the destination on receiving ORM from the source activates the SCH. For channel access, the data transmission occurs on slots allocated for a duration of  $\Delta$ DIFS. For the DIFS time, if the channel is idle, then the back-off scheme is applied based on the contention window size. A third-party handshake mechanism is employed for source, destination, and relay for both sm and nsm.

This protocol provides successful probability of transmission and collision. Hence, a high throughput, high packet delivery ratio, and minimized latency are achieved. The major shortcoming is that unsaturated network conditions are not considered. Channel fading conditions and other effects of signals such as capture effect are also not considered in performance analysis.

##### 4.7.2. RECV-MAC

Reliable and efficient cooperative (RECV) MAC is a novel protocol proposed to provide efficient communication with high throughput. The protocol provides better packet dropping rates (PDRs) and lowers delay for safety messages. CSMA/CA is used with random access as in IEEE 802.11p. Modified control messages are used to provide cooperative communication. Negative acknowledgement (NACK) and keen to help (KTH) messages are introduced for exchange of safety messages. Cooperative wave service advertisement (CWSA) and willing to involve (WTI) are introduced for non-safety messages. Packets transmit NACK; if the node hears broadcasts but does not receive a notification in short inter-frame space (SIFS), there is failed transmission. These packets are sent through cooperative transmission to improve the transmission reliability. After NACK, the nodes with improved transmission rate, SINR, and proper channel condition transmit KTH. Upon receiving KTH, an optimal helper node (having optimal SINR) can be chosen among neighboring nodes. Basic service set (BSS) is used for non-safety message transmission, with a period broadcasting WSA in the CCH interval. The receiver sends WTI, and when a neighbor node has better SINR and channel conditions, it sends CWSA to the sender. The node with optimal SINR is chosen as the optimal WAVE helper node; after hearing the

selector helper message (SHM), other nodes suspend cooperation with WTI. The optimal helper joins BSS, and thus, the sender transmits through the optimal helper to the receiver.

RECV MAC provides an optimal helper selection mechanism for reliable data transmission. However, the performance of the protocol in unsaturated conditions and in high vehicle density is not analyzed. The power consumption factor is not taken into consideration.

#### 4.7.3. OCA-MAC

The optimal cooperative ad hoc (OCA) MAC protocol was proposed to provide cooperative communication to improve transmission probability. The optimal cooperative node was chosen based on TDMA for successful transmission. A probabilistic model was designed to know the number of optimal cooperative nodes in each channel. The methods of cooperative forwarding and optimal node determination were used for reliable communication. The cooperative transmission mechanism was employed when the transmission rates go below the threshold. The node broadcast control frame periodically contains information about it such as node ID, neighbor node ID, and slot reserved. Choosing an optimal cooperative node is based on the distance calculated between the potential cooperative node and the mid-point between the sender and receiver, thereby increasing the network throughput. The node with a minimum value of distance calculated is chosen as the optimal cooperative node. To analyze the number of cooperative nodes, a probabilistic mathematical model was designed. In this model, the time slots are divided into free time slots, successful time slots, and failed time slots.

The protocol provides QoS in terms of successful transmission rate, transmission delay, and transmission in highway scenarios. Since the protocol does not consider the dynamic topology scenario, there might be performance degradation in dynamic scenarios.

#### 4.7.4. CT MAC

The proposed cooperative TDMA (CT) MAC uses a relay strategy to improve communication efficiency hampered by interference or jammers at RSUs. It uses a slot sharing scheme; the status information of each node is present in the MAC header. The slots are shared and prioritized by analyzing the slot state. Every time slot is partitioned into two CWs; DSRC specifies the payload for safety application and a slot information (SI) header. The carrier signal sensed by vehicles at each time slot is used to detect the state of the slot. This slot is again divided into three states: first, check the packet is correctly received; second, when the packet is received; and third, when no packet is received and SINR is low (channel is idle). The slot is prioritized by a value accumulated and received during a frame, containing the data generated by the neighbors. The category of the channel sensed that is idle, busy, or noisy is defined. An overview of CTMAC is shown in Figure 14.

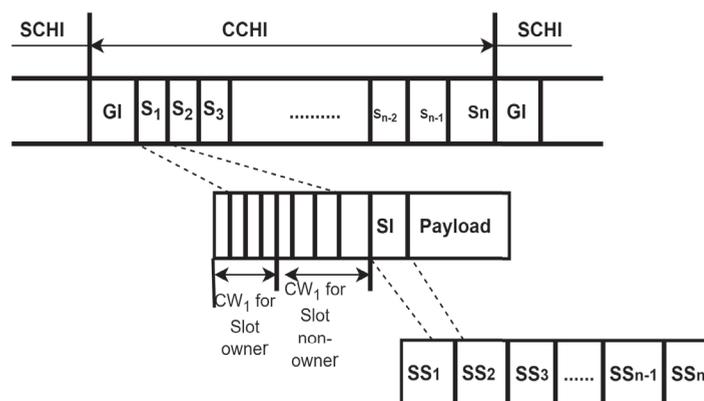


Figure 14. Overview of CT MAC [90].

The stability of the network is maintained by prioritizing those vehicles occupying the slot through a carrier sensing algorithm. The proposed work eliminates merger and access collision using the back-off method in the slots. Priority-based slot allocation avoids communication interference. However, the reliability and throughput metrics are not analyzed in dense and sparse scenarios.

#### 4.7.5. ST MAC

The spatio-temporal (ST) coordination-based MAC protocol for VANETs is designed to provide contention-free channel access for safety message exchange. It provides reliable and efficient data transmission for safe driving. Here, transmission occurs by line-of-collision graph based on a set-cover algorithm [93] so that vehicles transmit in the same slot but do not collide due to directional antennas and transmission power control. An optimization-based contention period scheme is proposed for vehicle registration on RSUs to minimize channel utilization. A hybrid MAC is designed by coordination of spatio-temporal characteristics based on PCF for registration of vehicles. This allocates slots and disseminates emergency messages from RSUs to vehicles based on DCF for safety and emergency messages via V2V communication by WPCF [94].

The proposed work provides better end-to-end delays and packet delivery ratio and less frame access delays. However, it does not handle non-safety-based applications, and the communication efficiency in highway scenarios for safe driving is not discussed. Since it is RSU centric, there is an overhead on RSU.

#### 4.8. Cluster-Based Single-Channel MAC Protocols

A cluster head (CH) is chosen from a group of vehicles that is responsible for channel access, allocation of time slots, and resource management. Clusters provide stable communication and extend the lifetime of communication links.

##### 4.8.1. PDMAC

Priority-based enhanced TDMA MAC was designed to prevent accidents via in-time delivery of time-critical safety messages and non-safety messages using the priority assignment technique. The protocol provides better clock synchronization, reduced message loss, latency, and improved throughput in the network. This protocol was based on an intra-cluster V2V model on two-directional highways and inter-cluster clock synchronization to reduce overhead and to improve channel utilization. It describes a three-tier priority assignment method for better delivery of warning messages estimating the type of message, severity level, and direction components. A single bit field called `Validate_timer` in the message header is used to check whether the timer is synchronized. The node's timer is considered synchronized when `Validate_timer=1` and, hence, validated. Conversely, the clock needs synchronization when `Validate_timer=0` and `Validate_timer` remains invalid for all other nodes on the network. `Validate_timer` is set the default value of 0 to make synchronization of clock mandatory for each and every node upon entering the highway [95]. The clustering mechanism of PD MAC in a bidirectional scenario is depicted in Figure 15.

The protocol's performance in urban or city scenarios (dense and highly mobile networks) shows less efficiency.

##### 4.8.2. Enhanced IEEE 802.15.4

The proposal of a V2R protocol based on the dynamic window algorithm (DWA) improves delay and throughput using a back-off scheme and the IEEE 802.15.4 standard employed for low power. Short-range communication is used in the PHY and MAC layer by CSMA/CA for contention access. A change in the binary exponent back-off (BEB) is implemented for improved performance. The frame structure known as super-frame is divided into two active parts containing a contention-free period (CFP) and a contention-based period (CBP), the inactive part, which is optional. The main objective of DWA is to

minimize the delay via an extended back-off period in an exponential back-off scheme. The nodes have queues and are arranged in the form of a cluster. The nodes with empty queues have low priority and nodes with full queues have high priority, i.e., the less frames in the node queue, the smaller the exponential window size. The size of the cluster is also taken into consideration: the more nodes in the cluster, the higher the probability of collision.

According to the simulation results, the DWA algorithm provides better results even when the cluster size is increased as it structures a balance between the nodes in the cluster and frames in the queues. Future enhancements include switching off at a threshold of different values between BEB and DWA and making the model energy-efficient by minimizing the power requirement with a collision avoidance scheme.

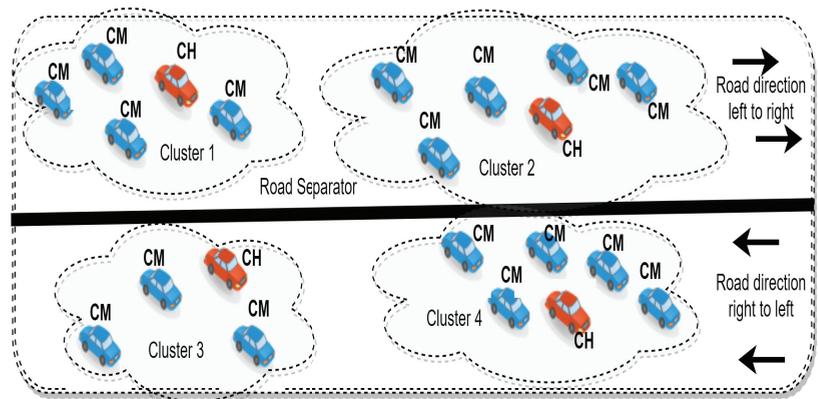


Figure 15. Clustering in PDMAC in the bidirectional traffic scenario [95].

#### 4.9. Cluster-Based Multi-Channel MAC Protocols

Basically, the cluster-based multi-channel MAC protocols are proposed to minimize channel contention, to increase network capacity by reusing resources, to provide fair access to channels within the cluster, and to efficiently control the network topology.

##### 4.9.1. CB MAC

Cluster-based MAC eliminates the hidden terminal problem for non-safety applications and efficient hand-over (shifting of data sessions from one base station to another). Each cluster head assigns bandwidths to the members of the cluster for efficient communication. As IEEE 802.11 is not cluster-based, the control packets are modified in CB-MAC with new control packets containing RTCF (request to cluster formation), ReTCL (registration to cluster), RCLM (request to cluster merging) [96]. For cluster member, a vehicle RTCF message is broadcast in the network; after any cluster head has received the message, it transmits ReTCL, which includes a cluster member ID(CM-ID), a cluster ID (CI-ID), and a cluster head address (CHA). Then, the cluster head updates the cluster member list and broadcasts it to all cluster members. If the new vehicle that wants to enter a cluster does not receive any ReTCL, it forms a new cluster and has to be the cluster head itself. If a CH does not receive any CTS when it has sends an RTS during the SIFS time interval, it assumes that the cluster member is out of the range of transmission and deletes it from the cluster member list. Then, again, the updated information is broadcast to all other members. A CH with a more significant number of cluster members becomes the CH.

This protocol is contention-free and provides high throughput, better PDR, and better resource utilization. When the number of vehicles increases rapidly, there might be chances of collision, leading to less throughput and a reduction in system performance.

#### 4.9.2. LMMC

A multi-channel MAC based on learning automata is designed for optimized utilization of channel access for VANET applications. The multi-channel scheme with the use of a radio transceiver and a smart model of learning automaton is used to learn about the cluster members' traffic parameters. Nodes are equipped with GPS, unlike other MACs that use CCH for safety applications and SCH for both safety and non-safety applications. The learning automaton model is implemented for the polling of cluster members to find out its parameter using the cluster head, with dynamic TDMA allocation of resources within the cluster [97].

The protocol provides better PDR for safety applications in real-time scenarios. It provides proper bandwidth and channel access utilization and, thus, increases throughput in the network. Still, there are not many analyses on QoS for non-safety applications. The protocol does not deal with best-effort, video, background, and other non-safety-based services.

#### 4.9.3. ACB MAC

It is a novel MAC protocol based on a clustering MAC with a blockchain framework providing reliable and secure communication. It uses the decentralized blockchain concept for vehicle authentication and registration before they communicate. The control packet of IEEE 802.11 was modified for handling hidden nodes, packet drop, and packet overloading on channel access. Cluster-based concepts provide a faster and efficient method of communication. The messages are divided into safety and non-safety messages and groups the vehicles as general vehicles and emergency vehicles. Priorities are set for general and emergency vehicles so that there is less delay on transmission of safety messages.

The protocol minimizes computational as well as storage overheads by efficient blockchain implementation. The protocol can be enhanced for different attacks and threats with reputation management, including more security features. A clustering based on the SDN architecture can be implemented in future enhancement for better network performance.

#### 4.10. Virtual Token Based

This approach allocates different time slots to vehicles, thereby minimizing collision and providing efficient channel access. The protocol also provides minimized propagation delay with high throughput.

#### Reliable MAC

The reliable MAC protocol scheme was proposed to avoid collision in high-density VANET scenarios via proper channel access. An adaptive byte-level HARQ was used for error control and to provide reliable communication. Via this methodology, the bit error rate and burst bit rate of the channel were calculated [98]. The frame error control redundancy size was checked, which was added to the MAC frame structure; along with the FEC bit, a checksum calculator was also added to the MAC frame. Frame segmentation and fragmentation were performed, and each frame consisted of MAC service data units. The virtual token approach performs the packet delivery mechanism (allocates each node with the specified channel access time) and forward node selection approach (selects efficient forwarders for message broadcasting) for on-time delivery of safety messages. The node that wants to transmit data initiates the virtual token formation containing the nodes present in its transmission range. The node initiator keeps track of nodes leaving and entering the token ring. An efficient forwarder node is selected that restricts other nodes to broadcast messages, thus avoiding collision.

The use of AB-HARQ shows better error recovery probability even upon increasing the channel error. The reliable MAC provides better throughput and packet delivery ratio. Hidden terminals, energy consumption, and communication overheads are not addressed or analyzed, turning out to be the major demerits.

#### 4.11. Random Selection Based

The time slot is accessed randomly; the channel is split up into frames and further divided into slots. This mechanism provides better channel utilization and minimizes collision. Some slots may be left idle and may be overloaded, leading to resource wastage and collision.

##### 4.11.1. OGC MAC

A novel OFDM MAC based on contention is proposed for reliable, efficient, fast, collision-free resource block allocation. The protocol reduces the use of contention windows (CWs) and improves resource allocation efficiency by proposing a new CCH architecture. Multi-carrier burst contention (MCBC) [99] inspired its creation, which reduces the size of CWs and reduces resource wastage by sequential channel allocation with the traditional TDMA approach. The bandwidth of CCH is divided into sub-channels by OFDM. A group contention strategy is proposed to overcome resource wastage along with a greedy approach. The RBs are chosen using the RB selection policy [100], where RBs are randomly selected from the groups formed from the subdivision of frame, to minimize collision when broadcasting a beacon to one-hop neighbor. A MAC announcement is initiated before a beacon is transmitted and embedded in the beacon. The RBs with a collision flag are considered collisions that occurred in the previous frame and are released. The nodes know the status of their two-hop neighbor and broadcast accordingly.

The protocol improves throughput by eliminating the hidden terminal problem and by avoiding merger and access collision. By proper resource utilization, resource wastage is minimized. The major disadvantage of this protocol is CW overload, causing delays in packet delivery. The network scalability in dense scenarios is not considered.

##### 4.11.2. Self-Sorting-Based MAC

This protocol is based on self-sorting the channel allocation based on TDMA. It provides collision-free transmission with reduced packet loss and delays in message delivery in even dense scenarios. The un-ordered random access is altered by ordered self-sorting-based access. Vehicles access channels in a queue via a TDMA-based mechanism. Nodes in the same queue compete to access the channel to minimize access collision via random channel access. The self-sorting process occurs in three steps: self-sorting, channel reservation, and data transmission [101]. Figure 16 depicts the mechanism of the protocol for sorting the message in queue, reservation of the channel, and the transmission of data.

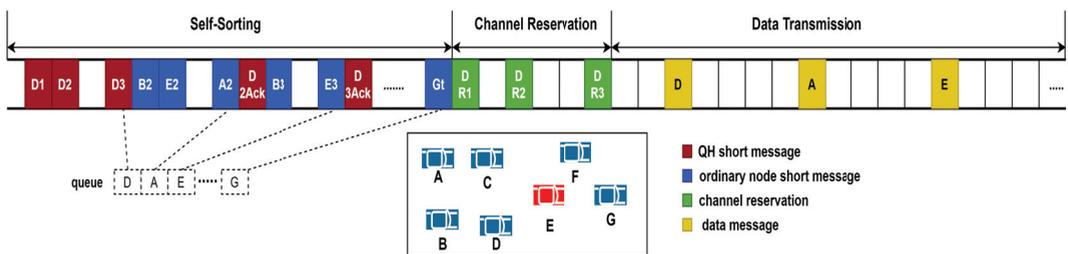


Figure 16. Mechanism of self-sorting [101].

As soon as the capacity of a queue reaches its threshold, the members or nodes of the queue start to transmit in the channel in the order in which they joined the queue. An M/G/1/∞ queuing model was used in the analysis for the computation of average service time, delay, and packet delivery ratio. Markov chain modeling was used to estimate the probability of queuing successfully.

The protocol overcomes the overhead of maintaining a schedule table, unlike other slotted protocols, as the nodes are self-sorted. It effectively handles delays and PDR.

However, adapting the dynamic scenarios, handling hidden terminals, and various other requirements of delay and PDR are still challenges. No analysis is given based on merging collision.

An extensive study of the MAC layer protocols for VANETs was conducted in the survey under various categories. These protocols aim to build an efficient mechanism for data dissemination. There are, however, numerous issues to be addressed in the future. The future directions include but are not limited to prioritizing safety messages for guaranteed reliability; ensuring fairness, in which a robust MAC protocol should also meet different QoS requirements, prioritizing services; handling network loads with dynamic channel access and proper channel utilization not limited to a single CCH; and minimizing interference by adopting a multi-channel access mechanism, where a centralized MAC protocol should be implemented under the premise of unreliable link and high vehicle mobility since RSUs and CHs can support direct multi-hop transmissions by adjusting to high dynamic networks. To conclude, the research direction includes designing an optimized MAC scheme providing QoS, adapting appropriate channel mechanisms based on the scenario (dense/sparse), and adapting clustering mechanisms for efficiency.

A chronological diagram of the above discussed MAC protocols is depicted in Figure 17, and a summary of the overall comparison is shown in Table 3.

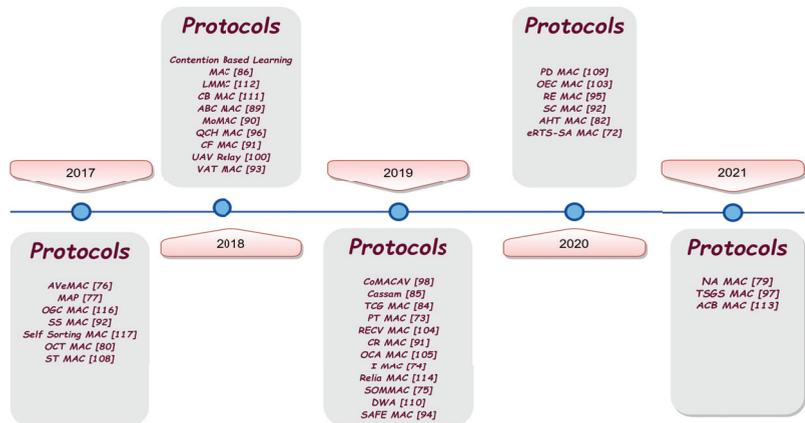


Figure 17. Chronological diagram of MAC protocols survey.

Table 3. Comparison of MAC protocols.

SR. No.	Protocols	Comparison	Objective	Methodology	Simulator	Classification	Pros	Cons
1	eRTS-SA [69]	VeMAC [53], RTS-SA [102]	Avoids interference, reduces overhead, and eliminates the hidden terminal problem	Micro-base station, physical interference, and SINR	SUMO	Hybrid and TDMA	High throughput and fair channel access.	Scalability and energy efficiency
2	PT MAC [70]	ADHOC MAC [47], Even Odd MAC [103]	Collision avoidance, minimized overheads, and high delivery ratio	Potential collision detection, potential collision prediction, and potential collision elimination	MATLAB, and SUMO	Prediction based, TDMA, contention free, and decentralized	Minimized delay and collision	Overhead and energy consumption
3	IMAC [104]	CF-MAC [83], DTB MAC [105]	Reduced collision, and load balancing and packet loss ratio minimized	CSMA	MATLAB R2018b	Distributed and contention free	Collision avoidance	Hidden terminal problem and overhead
4	SoMMAC [106]	SOFTMAC [57], VeMAC [53]	Reduced cost	Clustering, two-way scenario, and fixed RSUs	NS2	Multi-channel, unidirectional, contention based, distributed, cluster, and TDMA	Efficient handover, minimized collision, and high packet delivery	Power consumption and overhead
5	A VeMAC [71]	VeMAC [53]	Provide better performance in terms of access collision and channel utilization	TDMA, random selection of time slots, and one-hop and two-hop neighbor division	Analysis	Reservation based, multi-channel, and TDMA	Eliminates collision and handles unbalanced traffic	Hidden-node problem and throughput not analysed
6	MAP [72]	PLNC [107]	Handles collision and hidden node problem	Multi-hop data dissemination, two-phase transmission: multiple access phase and broadcast phase	Theoretical analysis	Decentralized, multi-hop, location based, and priority based	Reliable and avoids collision	Merging collision and throughput not analysed

Table 3. Cont.

SR. No.	Protocols	Comparison	Objective	Methodology	Simulator	Classification	Pros	Cons
7	NA MAC [74]	HC MAC [108], VeMAC [53]	Efficient slot allocation and hidden terminal problem	Three-way handshake with TDMA and CSMA/CA	NS3	Distributed single channel	Reliable broadcast of safety message	Broadcast storm problem
8	OCT MAC [75]	TDMA and CDMA MAC, 802.11 Model [109]	Improves network throughput and channel quality	Optical channel DC model and CDMA, time synchronization (1PPS signal), and FCS at the physical layer	NS3	TDMA, SNR, distributed	High throughput and minimized delay	Performance degradation in dynamic environment
9	AHT MAC [77]	Ve MAC [53], SS MAC [84], OGC MAC [100]	Resource wastage and overhead minimization	Two-way handshake, adaptive resource reservation, request conflict resolution, and synchronized switching	NS3 and SUMO	TDMA, multi-channel, and hybrid	Enhanced throughput and less delay	Overhead and packet loss
10	SC MAC [110]	TDMA and CSMA/CA based [59], BB MAC [111], DBCC [112]	Scalable, low latency, and high PDR	Black burst-based slot reservation and spatio-temporal coordination, and periodic random back-off scheme	Analysis	Hybrid, contention based, TDMA, and distributed	Alleviates collision and high PDR	Transmission delay
11	TCGMAC [78]	VeMAC [53], MoMAC [82]	Minimizes transmission collision and increases throughput	Frame divided into TP (transmission period) and CP (CSMA period), and game theory	MATLAB	TDMA + CDMA based and distributed	Collision alleviation	High power consumption and overhead
12	Cassam [79]	IEEE 802.11-OCB (CSMA/CA) [113]	Guaranteed safety application in real-time and hostile environments	Slotted 1 persistent	SUMO	Context aware, decentralized, and CSMA/CA	Minimized delay and enhanced throughput	Power consumption not analysed

Table 3. Cont.

SR. No.	Protocols	Comparison	Objective	Methodology	Simulator	Classification	Pros	Cons
13	Contention Based Learning MAC for Broadcast [114]	IEEE 802.11p [115]	Scalability, collision avoidance, and bandwidth wastage reduction	MAC+PHY, RL model, and discrete time stochastic control	OMINET++ and SUMO	Contention based and discrete time synchronization	High PDR and throughput	Fair-channel access and latency
14	ABC MAC [116]	LIMERIC [117], Conventional IEEE 802.11p	Avoid blocking of control channel and fair channel access	Heuristic greedy beacon congestion control	SUMO	TDMA-based broadcast MAC and distributed	High transmission ratio and avoids collision	No emphasis on non-safety message and throughput degradation in the dense scenario
15	MoMAC [82]	TDMA MAC [118], ADHOC, VeMAC [53]	Eliminates collision, handles hidden terminal, and delay-tolerant transmission	Distributed slot access, collision detection, and TDMA	SUMO	TDMA, distributed, and contention free	High reception rate and minimized delay	Cannot handle dynamic traffic
16	CF MAC [83]	DTMAC [119]	Collision-free transmission and QoS enhancement	Dynamic self-configured TDMA and platooning	MATLAB R2017a	Contention free and TDMA	Reliability and collision avoidance	Delay and performance degrades in dense scenarios
17	SS MAC [84]	Aggressive MAC and conservative MAC scheme	Reliable broadcast of safety messages	Distributed time slot sharing algorithm and random index first fit (RIFF) algorithm (heuristic packaging method)	MATLAB and SUMO	Periodical and distributed	Scalable, reliable, and minimized delay	Overhead and handling dynamic scenarios
18	VAT MAC [85]	VeMAC [53], ATSA [118]	Efficient broadcast of safety messages	Estimation of nodes, determination of leaving node, predicting newly entering node, and frame optimization	NS-2.35 and SUMO	Contention free, dynamic access, and time synchronized	Enhanced scalability and throughput	No proper time synchronization mechanism and delays

Table 3. Cont.

SR. No.	Protocols	Comparison	Objective	Methodology	Simulator	Classification	Pros	Cons
19	SAFE MAC [86]	standard 802.11 [120]	Fair channel access (V2I)	Batch selection and residence time calculation	Numerical analysis	Contention-based, CSMA/CA, and time synchronized	Minimized delay	Fairness issue not handled for V2V and V2D
20	ReMAC [87]	VeMAC [53], CFR MAC [121], VAT MAC [75]	Efficient channel usage	DSRC standard, and CCH and SCH organization with joining network channel	OMNET++	Hybrid, multi-channel, and CSMA + FDMA	Minimized delay and high throughput	Communication overhead and energy consumption
21	QCH MAC [88]	CAC [122], EDCA-DCRRSU [123]	Delay constraint safety applications	EDCA combined with TDMA and two access periods: transmission period and reservation period	SUMO and NS 3.26	Hybrid MAC, distributed, and TDMA+EDCA	Enhanced QoS	Hidden terminal problem and throughput can still be maximized
22	TSGS MAC [124]	Random sending	Minimizes collision and reduces overlapping connections	Scheduling strategy based on connection level	OMNET++ and SUMO	Centralized multi-channel and contention-based	Less delay and high PDR	overhead and hidden terminal
23	CoMACAV [90]	CAHMAC [55], SR MAC [125], and VeMAC [60]	Increasing throughput by selecting optimal relay using the Markov chain model	Three modes of transmission: DT, CR, and MHR	MATLAB	Contention based, cooperative, SNIR, and multi-hop	High throughput	Delays
24	CR MAC [89]	ADHOC MAC (CAH MAC) [53] and Relay MAC [126]	A novel cooperative relay procedure	Three modes of data transmission: DT, CR, and MR; modification of control packets of IEEE 802.11	MATLAB	Cooperative, contention based, and multi-hop	Enhanced throughput	Delays

Table 3. Cont.

SR. No.	Protocols	Comparison	Objective	Methodology	Simulator	Classification	Pros	Cons
25	UAV Relay in VANET with RL [127]	Q-learning-based scheme [128]	Reduces the bit rate error and jamming resistance for UAV	Relay game using Nash equilibrium (NE), hotbooting PHC (policy hill climbing), path loss, shadowing, and Rayleigh fading	SUMO	Smart jammer, hotbooting PHC based, and RL based	Minimized interference	High computational overhead
26	OEC MAC [129]	RECV MAC [130]	Optimal relay, appropriate transmission mode, and minimized delay	OFDMA, FSM with UML, and SINR	MATLAB	OFDMA, cooperative, SINR, and distributed	High throughput, PDR, and less delay	Performance degradation for unsaturated network
27	RECV MAC [130]	Traditional MAC	Reliable and efficient communication, and less packet dropping rate and delays	Markov model analysis, optimal helper, and use of FSM and UML	MATLAB	Random access, CSMA/CA, and cooperative	Reliable	Performance degradation in very high density and unsaturated scenarios
28	OCA MAC [131]	VeMAC [53], CAH MAC [55]	Improves probability of successful transmission rate	Cooperative forwarding; selecting optimal cooperative node; and time slots divided: free, successful and failed	NS2	Cooperative TDMA based, probabilistic, multi-channel, centralized, and contention free	QoS in terms of throughput, delay, and PDR achieved	Performance of dynamic scenarios not analysed
29	CT MAC [132]	VeMAC [53], DTMAC [119]	Solve hidden terminal and collision problem.	Channel scheduling and back-off algorithm	Analysis	Cooperative and distributed	Eliminate collision	Throughput metrics not analysed in dense/sparse scenarios

Table 3. Cont.

SR. No.	Protocols	Comparison	Objective	Methodology	Simulator	Classification	Pros	Cons
30	ST MAC [94]	LMA [133], DMMAC [134], and EDCA	Improves channel efficiency, reduces delay, and optimizes the contention period & spatio-temporal transmission scheduling technique and R-ALOHA	Nakagami3 Radio Model and SUMO	Hybrid (PCF+DCF), contention free, and spatio-temporal coordination based	Minimized delay and high PDR	Overhead on RSU	
31	PD MAC [95]	DMC MAC [135,136]	Reliability and on-time delivery of time-critical messages to reduce accidents	Inter- and Intra-cluster clock synchronization, and three-tier priority assignment	VANET Toolbox	Priority, cluster based, and TDMA	Minimized delay	Performance degradation in urban scenarios
32	Enhancement of IEEE 802.15.4 MAC(DWA) [137]	BEB [138]	Smart back-off algorithm that improves delay and throughput	BEB and improves CSMA/CA used in traditional IEEE 802.15.4	Analysis	IoT based, cluster, and dynamic window	Improved throughput and minimized delay	Energy utilization
33	CB MAC [96]	TDMA-based MAC [118] and RTS/CTS mechanism	Reduces contention, manages the network topology efficiently, and solves the hidden terminal problem	Formation of clusters, RTS/CTS for non safety messages, and CH and CM handshake for safety message transmission	MATLAB	Cluster, RTS/CTS, and handshake	High throughput and PDR	Cannot handle high vehicle densities
34	LMMC [97]	TCMAC [139], ASAS (Adaptive TDMA) [140], and ETCM [141]	Addressing spectrum scarcity problem and optimal channel utilization	Learning Automata for clustered VANET and probability calculation	NS 2.35 with SUMO	Cluster-based multi-channel MAC, dynamic TDMA, and CSMA/CA	Improved PDR	Does not prioritize messages (non-safety)

Table 3. Cont.

SR. No.	Protocols	Comparison	Objective	Methodology	Simulator	Classification	Pros	Cons
35	ACB MAC [142]	Traditional CB MAC	Collision free and authenticated reliable communication	Cluster based, blockchain framework, and prioritizing safety message	MATLAB	Cluster-based and multi-channel MAC	Minimized delay for safety messages and secure	Energy consumption and less throughput
36	Relia MAC [98]	DTB MAC [105] and oppcast MAC [143]	Timely and reliable transmission of safety data, cope with dynamic mobility pattern, and error prone communication	MAC fragmentation and segmentation, encoding FEC, and checksum calculator for ARQ	NS2	Virtual token, distributed, sequential, and dynamic environment	High throughput and PDR	Overhead and energy
37	OGC MAC [100]	VeMAC [53], MCBC [99]	Proper resource allocation and channel access, and reliability	Random selection, contention strategy, and probability for collision free beacon broadcasting	SUMO	Parallel contention mechanism and OFDM	High throughput	Scalability issue and delays due to CW overload.
38	Self-Sorting-Based MAC [101]	Hassen's Model, DMIMAC [134], and VeMAC [53]	Minimizes channel collision and minimizes overhead	Data sent periodically, Self sorting by Markov chain, channel reservation, and probability of successful queuing is calculated	SUMO	TDMA, contention free, random access, and Distributed	Reduce packet loss and delay, and high throughput	Handling dynamic scenario and hidden terminal problem

## 5. Future Scope, Open Challenges, and Research Direction

Being one of the promising technologies for ITS, VANETs provide safety, efficient driving, infotainment services, pollution-free driving, and a pleasant environment for drivers. To provide these services in smart vehicles based on IoT, Internet of Vehicles (IoVs) are designed. Handling real-time events efficiently and reliably for safety applications is a foremost challenge. Many challenges have already been overcome by using new technologies such as edge cloud-computing (ECC) for resource virtualization, IoV for smart communication, SDN for network virtualization, and clustering for efficient communication. However, problems still persist and give a thorough layout for future scopes and research directions alike such as designing a low latency protocol for real-time safety applications. Providing a coherent connection among vehicles is essential to maintaining efficient and reliable communication of vehicles but is a challenge due to the dynamic environment of VANETs. Nonetheless, providing high bandwidth is necessary for streaming and other non-safety 3D-based, high-dimensional applications such as navigation systems, reading of maps, games, video streaming, etc. However, bandwidth is mostly impeded due to the dynamic VANET topology. Hence, designing an efficient MAC for VANETs is necessary for both safety-critical and non-safety critical applications.

Congestion control in the network is an essential factor for providing collision-free communication and high network throughput since a long CSMA/CA has been used for congestion control and, in recent research, many modifications have also proposed efficient algorithms for congestion control. Improved channel utilization with continuous and alternate channel access and broadcast safety messages effectively is yet another challenge and future scope. Hidden and exposed node problems should be dealt with efficiently to improve throughput, scalability, and network performance. Time-synchronization among vehicles is also an important issue that needs to be maintained to provide collision-free transmission in multi-channel scenarios that provide proper time slots to the nodes. Additionally, it offers optimized performance and load-balancing in terms of channel access. The minimization of communication overhead opens discussion on the implementation of centralized and distributed protocols. Centralized protocols suffer from overhead, latency, scalability, and flexibility, whereas the decentralized approach is not easily maintained and has a high cost.

## 6. Conclusions

The objective of this survey was to obtain insight into various MAC protocols designed for VANETs. This paper presents a brief introduction to VANETs along with its challenges and applications. The protocols are classified into distributed, centralized, cluster-based, random-access, and virtual token-based and sub-classified into single-channel and multi-channel MAC protocols. The identified primary challenges in the design of MAC protocol are latency, dynamic topology, mobility of nodes, reliability, and bandwidth utilization. This paper also briefly elaborates on various DSRC-based and cellular-based networks, giving a gist of the ITS-G5 standard, LTE, 5G, and C-V2X communication. Furthermore, Table 3 presents a structured, detailed review of recent MAC protocols along with their mechanisms, objectives, comparisons, and classifications.

An efficient MAC design is of utmost importance to providing delay-intolerant and reliable messaging. Data packets are prioritized based on safety and non-safety applications and disseminated based on priority. With multi-channel protocols, multiple channels can handle interference caused by channel access, resulting in better channel utilization. A multi-channel clustering mechanism with dynamic interval-based channel access is an open challenge for MAC protocols to provide QoS and to improve network performance. Technologies such as SDN, which separates the data plane from the control plane, have also emerged to eliminate traffic congestion. Edge-computing vehicles, VANETs based on the Internet of Things, and cloud-based VANETs are all emerging technologies for reliable and efficient transportation. Multi-transceivers, multi-channel operations, and multiple inputs and outputs (MIMO) are some methods for enhancing the network throughput. Future

work includes experimental analyses of the MAC protocols in VANET in terms of QoS metrics such as throughput, packet reception ratio, energy consumption, and performance improvements using an efficient MAC design.

**Author Contributions:** This work was completed with contributions from all the authors. conceptualization, L.H. and A.K.; methodology, L.H. and A.K.; software, L.H., B.P.N. and A.K.; validation, formal analysis, data curation and writing—original draft preparation, L.H., A.K., and G.G.M.N.A.; writing—review and editing, L.H., A.K., G.G.M.N.A. and P.H.J.C.; visualization, L.H., and B.P.N.; funding acquisition, P.H.J.C. All authors did edit, review and improve the manuscript. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Not applicable, the study does not report any data.

**Conflicts of Interest:** The authors declare no conflict of interest.

**Abbreviations**

The following abbreviations are used in this manuscript:

Acronym	Description	Acronym	Description
3GPP	Third Generation Partnership Project	AB HARQ	Adaptive Byte-Level Hybrid Automatic Repeat reQuest
ABC MAC	Adaptive Beacon Control MAC	AC	Access Category
ACB MAC	Authentication Cluster-Based MAC	ACFM	Adaptive Collision-Free MAC
ACK	Acknowledgment	AHT MAC	Adaptive High-Throughput MAC
ASAS	Adaptive TDMA Slot Assignment Strategy	AVeMAC	Adaptive Vehicular MAC
BB MAC	Black Burst MAC	BEB	Binary Exponent Back-Off
BER	Bit Error Rate	BFP	Broadcast Feedback Phase
BSS	Basic Service Set	BT	Broadcast Table
BTMA	Busy Tone Multiple Access	CAC MAC	Cellular Automata to Form CAR MAC
CAH MAC	Cooperative Ad Hoc MAC	CAM	Cooperation Acceptance Message
CAP	Contention Access Phase	CaSSaM	Context Aware System for Safety Messages
CB MAC	Cluster-Based MAC	CBCH	Contention-Based Sub-Channel
CBP	Contention-Based Period	CBR	Channel Busy Rate
CCH	Control Channel	CDMA	Code Division Multiple Access
CF MAC	Collision-Free MAC	CFP	Contention-Free Phase
CFP	Contention-Free Period	CFR MAC	Collision-Free Reservation MAC
CH	Cluster Head	CI	Control Interval
CoMACAV	Cooperative MAC for Autonomous Vehicle	CR	Cooperative Relaying
CRM	Cooperation Request Message	CRMAC	Cooperative Relay-Based MAC
CRP	Channel Request Packet	CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance

Acronym	Description	Acronym	Description
CT MAC	Cooperative TDMA MAC	CTS	Clear To Send
C-V2X	Cellular-Vehicle-to-Everything	CW	Contention Window
CWSA	Cooperative Wave Service Advertisement	CWSA	Cooperative Wave Service Advertisement
D2D	Device-to-Device	DBCC	Distributed Beacon Congestion Control
DBRA	Distributed Beacon Rate Adaptive	DCF	Distributed Coordination Function
DIFS	Dynamic Inter-Frame Spacing	DMMAC	Dedicated Multi-Channel MAC
DNS	Direct Neighbor Set	DSRC	Dedicated Short-Range Communication
DT	Direct Transmission	DT MAC	Distributed TDMA MAC
DTA	Data Transmission Acceptance	DTB MAC	Dynamic Token-Based MAC
DTR	Data Transmission Request	DTSS	Distributed Time Slot Sharing
DWA	Dynamic-Window Algorithm	EDCA	Enhanced Distributed Channel Access
eRTS-SA	Request Transmission Split-Slotted ALOHA	ETCM	Enhanced TDMA Cluster-Based MAC
ETSI	European Telecommunications Standards Institute	FCC	Federal Communication Commission
FDMA	Frequency Division Multiple Access	FI	Frame Information
GPS	Global Positioning System	HBCH	Heart-Beat Channel
HC MAC	Hybrid CSMA MAC	HER MAC	Hybrid Efficient and Reliable MAC
I MAC	Improvised MAC	IBT	Initial Broadcast Table
INS	Indirect Neighbor Set	IR	Interference Range
ITS	Intelligent Transportation System	JNCH	Joining Network Channel
KTH	Keen To Help	LIMERIC	Linear Message Rate Control Algorithm
LMA	Location and Mobile Aware MAC	LMMC	Learning Automata Multi-Channel MAC
LTE	Long-Term Evolution	MAC	Media Access Control
MACA	Multiple Access with Collision Avoidance	MACAW	Multiple Access with Collision Avoidance Wireless
MANET	Mobile Ad Hoc Network	MAP	Medium Access for PLNC (Physical Layer Network Coding)
mBS	Micro Base Station	MCBC	Multi-Carrier Burst Contention
MC-MAC	Multi-Code Media Access Control	MEC	Multi-Access Edge Computing
MHR	Multi-Hop Relaying	MoMAC	Mobility-Aware MAC

Acronym	Description	Acronym	Description
NA MAC	Neighbor Association-Based MAC	NACK	Non-Acknowledgement
NAV	Network Allocation Vector	NEMO	Network Mobility
NC-MAC	Network Coding MAC	OBU	On-Board Unit
OCA MAC	Optimal Cooperative Ad Hoc MAC	OCC	Optimal Orthogonal Codes
OCT MAC	Optical CDMA with TDMA MAC	OEC MAC	OFDMA-Based Efficient Cooperative MAC
OFDMA	Orthogonal Frequency Division Multiplexing Access	OGC MAC	OFDM-Based Group Contention MAC
OppCast	Opportunistic Broadcast	OR	Offer to Relay
ORM	Optimal Relay Message	PCF	Point-Coordination Function
PDMAC	Priority-Based Enhanced TDMA MAC	ProSe	Proximity Service
PHC	Policy Hill Climbing	PLNC	Physical Layer Network Coding
PPS	Pulse Per Second	PTMAC	Prediction-Based TDMA MAC
QCH MAC	QoS-Aware Centralized Hybrid	QoS	Quality of Service
RACK	Relay Acknowledgement	RCLM	Request to Cluster Merging
RCR	Relay Clear to Relay	RECV MAC	Reliable and Efficient Cooperative MAC
Relia MAC	Reliable MAC	ReMAC	Reliable MAC
ReTCL	Registration To CLuster	RIFF	Random Index Fit First
RP	Reservation Period	RRR	Relay Ready to Relay
RRTS	Relay Request To Send	RSU	Road-Side Unit
RSUHB	RSU Heart-Beat	RTCF	Request To Cluster Formation
RTS	Request To Send	SAFE MAC	Speed Aware Fairness-Enabled MAC
SB-SPS	Sensing Based Semi-Persistent Scheduling	SCCH	Scheduling Channel
SCH	Service Channel	SCHI	Service Channel Interval
SCMAC	Slotted Contention-Based MAC	SDMA	Spatial Division Multiple Access
SDN	Software Defined Network	SHM	Selector Helper Message
SIC	Successive Interference Cancellation	SIFS	Short Inter-Frame Space
SINR	Signal to Interference Noise Ratio	SNR	Signal-to-Noise Ratio
SOFT MAC	Space-Orthogonal Frequency Time MAC	SOMMAC	Self-Organizing Multi-Channel MAC
SR MAC	Simple Relay-Enabled MAC	SRBs	Service Resource Blocks
SS MAC	Slot Sharing MAC	ST MAC	spatio-temporal Coordination-Based MAC
STC	spatio-temporal Coordination	TC MAC	TDMA Cluster-Based MAC

Acronym	Description	Acronym	Description
TCG MAC	TDMA with CSMA MAC	TDMA	Time Division Multiple Access
TP	Transmission Period	TR	Transmission Range
TSGS MAC	Transmission Scheduling Greedy Search	UAV MAC	Unmanned Aerial Vehicle MAC
V2I	Vehicle-to-Infrastructure Communication	V2V	Vehicle-to-Vehicle Communication
VANET	Vehicular Ad Hoc Network	VAT MAC	Novel Adaptive TDMA MAC
VCAT	Vehicle Channel Allocation Table	VHB	Vehicle Heart-Beat
VoLTE	Voice-Over Long-Term Evolution	VSIP	Vehicular Specific Packet type
WAVE	Wireless Access in Vehicular Environment	WPCF	WAVE PCF
WSA	WAVE Service Advertisement	WTI	Willing To Involve
WTS	Want To Send		

## References

- Maalej, Y.; Abderrahim, A.; Guizani, M.; Hamdaoui, B.; Balti, E. Advanced Activity-Aware Multi-channel Operations 1609.4 in VANETs for Vehicular Clouds. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016; Volume 4, pp. 142–149.
- Kakarla, J.; Sathya, S.S. A Survey and Qualitative Analysis of Multi-channel MAC Protocols for VANET. *Int. J. Comput. Appl.* **2012**, *38*, 0975–8887. [[CrossRef](#)]
- AlMheiri, M.S.; AlQamzi, S.H. MANETs and VANETs Clustering Algorithms: A Survey. In Proceedings of the 8th IEEE GCC Conference and Exhibition, Muscat, Oman, 1–4 February 2015; pp. 1–4.
- Gillani, S.; Shah, A.P.; Qayyum, A.; Hasbullah, B.H. MAC Layer Challenges and Proposed Protocols for Vehicular Adhoc Networks. *Adv. Intell. Syst. Comput.* **2015**, *306*, 3–13. [[CrossRef](#)]
- Sheikh, S.M.; Liang, J. A Comprehensive Survey on VANET Security Services in Traffic Management System. *Wirel. Commun. Mob. Comput.* **2019**, *2019*, 2423915. [[CrossRef](#)]
- Sheikh, S.M.; Liang, J.; Wang, W. A Survey of Security Services, Attacks, and Applications for Vehicular Ad Hoc Networks (VANETs). *Sensors* **2019**, *19*, 3589. [[CrossRef](#)] [[PubMed](#)]
- Kelarestaghi, B.K.; Mahsa Foruhandeh, M.; Heaslip, K.; Gerdes, R. Survey on Vehicular Ad Hoc Networks and Its Access Technologies Security Vulnerabilities and Countermeasures. *arXiv* **2019**, arxiv:1903.01541.
- Dhamgaye, A.; Chavhan, N. Survey on security challenges in VANET. *Int. J. Comput. Sci. Netw.* **2013**, *2*, 2277–5420.
- Fuentes, D.J.; Manzano, G.L.; González-Tablas, I.A.; Blasco, J. Security models in vehicular ad-hoc networks: A survey. *IETE Tech. Rev.* **2014**, *31*, 47–64. [[CrossRef](#)]
- Hadded, M.; Muhlethaler, P.; Laouiti, A.; Zagrouba, R.; Saidane, A.L. TDMA-based MAC protocols for vehicular ad hoc networks: A survey, qualitative analysis, and open research issues. *IEEE Commun. Surv. Tutor* **2015**, *17*, 2461–2492. [[CrossRef](#)]
- Mohammed, A.M.; Al-Daraiseh, A.A. Toward secure vehicular adhoc networks: A survey. *IETE Tech. Rev.* **2012**, *29*, 80–89.
- Choi, J.; Marojevic, V.; Carl, B.; Dietrich, B.C.; Reed, H.J.; Ahn, S. Survey of Spectrum Regulation for Intelligent Transportation Systems. *IEEE Access* **2020**, *8*, 140145–140160. [[CrossRef](#)]
- Nguyen, V.; Kim, T.T.O.; Pham, C.; Oo, Z.T.; Tran, H.T.; Hong, S.C.; Huh, N.E. A survey on adaptive multi-channel MAC protocols in VANETs using Markov models. *IEEE Access* **2018**, *6*, 16493–16514. [[CrossRef](#)]
- Khan, A.U.; Lee, S.S. Multi-Layer Problems and Solutions in VANETs: A Review. *Electronics* **2019**, *8*, 204. [[CrossRef](#)]
- Teixeira, F.A.; Silva, V.F.; Leoni, J.L.; Macedo, D.F.; Nogueira, J.M. Vehicular networks using the IEEE 802.11p standard: An experimental analysis. *Veh. Commun.* **2014**, *1*, 91–96 [[CrossRef](#)]
- Kiela, K.; Barzdenas, V.; Jurgo, M.; Macaitis, V.; Rafanavičius, J.; Vasjanov, A.; Klodovscikov, L.; Navickas, R. Review of V2X–IoT Standards and Frameworks for ITS Applications. *Appl. Sci.* **2020**, *10*, 4314. [[CrossRef](#)]
- Kim, J.W.; Kim, J.W.; Jeon, D.K. A Cooperative Communication Protocol for QoS Provisioning in IEEE 802.11p/Wave Vehicular Networks. *Sensors* **2018**, *18*, 3622. [[CrossRef](#)] [[PubMed](#)]
- Yao, Y.; Rao, L.; Liu, X. Performance and Reliability Analysis of IEEE 802.11p Safety Communication in a Highway Environment. *IEEE Trans. Veh. Technol.* **2013**, *62*, 4198–4212. [[CrossRef](#)]
- Nardini, G.; Viridis, A.; Campolo, C.; Molinaro, A.; Stea, G. Cellular-V2X Communications for Platooning: Design and Evaluation. *Sensors* **2018**, *18*, 1527. [[CrossRef](#)] [[PubMed](#)]

20. Bai, F.; Krishnan, H. Reliability Analysis of DSRC Wireless Communication for Vehicle Safety Applications. In Proceedings of the IEEE Conference on Intelligent Transportation Systems—ITSC, Toronto, ON, Canada, 17–20 September 2006; pp. 355–362. [CrossRef]
21. Gast, S.M. *802.11 Wireless Networks: The Definitive Guide*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2002; ISBN 9780596100520. Available online: <http://196.223.158.148/bitstream/handle/123456789/2499/E302.pdf?sequence=1&isAllowed=y> (accessed on 8 November 2021).
22. Reddy, G.R.; Ramanathan, R. An Empirical study on MAC layer in IEEE 802.11p/ WAVE based Vehicular Ad hoc Networks. In Proceedings of the 8th International Conference On Advances in Computing & Communications (ICACC), Kochi, India, 13–15 September 2018; Volume 143, pp. 720–727. [CrossRef]
23. Liu, W.; He, X.; Huang, Z.; Ji, Y. Transmission Capacity Characterization in VANETs with Enhanced Distributed Channel Access. *Electronics* **2019**, *8*, 340. [CrossRef]
24. 3GPP Release 14. Available online: <https://www.3gpp.org/release-14> (accessed on 1 September 2021).
25. Chen, S.; Hu, J.; Shi, Y.; Zhao, L.; Li, W. A Vision of C-V2X: Technologies, Field Testing, and Challenges With Chinese Development. *IEEE IoT J.* **2020**, *7*, 3872–3881. [CrossRef]
26. Katranaras, E.; Dillinger, M.; Abbas, T.; Theillaud, R.; Calvo, J.L.; Zang, Y. Standardization and Regulation. In *Cellular V2X for Connected Automated Driving*; John Wiley Sons, Ltd.: Hoboken, NJ, USA, 2021; Chapter 3, pp. 63–90. [CrossRef]
27. McCarthy, B.; Burbano-Abril, A.; Rangel-Licea, V.; O'Driscoll, A. OpenCV2X: Modelling of the V2X Cellular Sidelink and Performance Evaluation for Aperiodic Traffic. *arXiv* **2021**, arXiv:2103.13212.
28. Noor-A-Rahim, M.; Liu, Z.; Lee, H.; Ali, G.G.M.N.; Pesch, D.; Xiao, P. A Survey on Resource Allocation in Vehicular Networks. *IEEE Trans. Intell. Transp. Syst.* **2020**, *2020*, 3019322. [CrossRef]
29. Mosavat-Jahromi, H.; Li, Y.; Cai, L.; Lu, L. NC-MAC: Network Coding-based Distributed MAC Protocol for Reliable Beacon Broadcasting in V2X. In Proceedings of the GLOBECOM 2020—IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; pp. 1–6. [CrossRef]
30. Saifuddin, M.; Zaman, M.; Toghi, B.; Fallah, Y.P.; Rao, J. Performance Analysis of Cellular-V2X with Adaptive and Selective Power Control. *arXiv* **2020**, arXiv:2008.03453.
31. Abdel Hakeem, S.A.; Hady, A.A.; Kim, H. 5G-V2X: Standardization, architecture, use cases, network-slicing, and edge-computing. *Wirel. Netw.* **2020**, *26*, 6015–6041. [CrossRef]
32. Bazzi, A.; Berthet, A.O.; Campolo, C.; Masini, B.M.; Molinaro, A.; Zanella, A. On the Design of Sidelink for Cellular V2X: A Literature Review and Outlook for Future. *IEEE Access* **2021**, *9*, 97953–97980. [CrossRef]
33. Boban, M.; Kousaridas, A.; Manolakis, K.; Eichinger, J.; Xu, W. Use Cases, Requirements, and Design Considerations for 5G V2X. *arXiv* **2017**, arXiv:1712.01754.
34. Garcia, M.H.C.; Molina-Galan, A.; Boban, M.; Gozalvez, J.; Coll-Perales, B.; Sahin, T.; Kousaridas, A. A Tutorial on 5G NR V2X Communications. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1972–2026. [CrossRef]
35. Tassi, A.; Egan, M.; Piechocki, R.J.; Nix, A. Modeling and Design of Millimeter-Wave Networks for Highway Vehicular Communication. *IEEE Trans. Veh. Technol.* **2017**, *66*, 10676–10691. [CrossRef]
36. Singh, P.K.; Nandi, S.K.; Nandi, S. A tutorial survey on vehicular communication state of the art, and future research directions. *Veh. Commun.* **2019**, *18*, 100164. [CrossRef]
37. Rebei, A.; Hadded, M.; Touati, H.; Boukhalfa, F.; Muhlethaler, P. MAC-aware Routing Protocols for Vehicular Ad Hoc Networks: A Survey. In Proceedings of the International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 17–19 September 2020; pp. 1–6. [CrossRef]
38. Shukla, N.; Katiyar, P.; Devgan, N.; Kumar, N.; Dinker, G.A. Comparison of the MAC Protocols in Vehicular Ad-Hoc Network. *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.* **2014**, *3*, 6709.
39. Jayaraj, V.; Hemanth, C.; Sangeetha, R.G. A survey on hybrid MAC protocols for vehicular ad-hoc networks. *Veh. Commun.* **2016**, *6*, 29–36. [CrossRef]
40. Huang, J.; Li, Q.; Zhong, S.; Liu, L.; Zhong, P.; Wang, J.; Ye, J. Synthesizing Existing CSMA and TDMA Based MAC Protocols for VANETs. *Sensors* **2017**, *17*, 338. [CrossRef] [PubMed]
41. Menouar, H.; Filali, F.; Lenardi, M.E. A Survey and Qualitative Analysis of MAC Protocols for Vehicular Ad Hoc Networks. *IEEE Wirel. Commun.* **2006**, *6*, 1536–1284. [CrossRef]
42. Jurdak, R.; Lopes, V.C.; Baldi, P. A Survey, Classification and Comparative Analysis of Medium Access Control Protocols for Ad Hoc Networks. *IEEE Commun. Surv.* **2004**, *6*, 2–16. [CrossRef]
43. Karn, P. MACA-A New Channel Access Method for Packet Radio. In Proceedings of the ARRL/CRRL Amateur Radio 9th Computer Networking Conference, London, ON, Canada, 22 September 1990; pp. 134–140.
44. Bargrodi, R.; Demers, A.; Shenker, S.; Zhang, L. MACAW: A Media Access Protocol for Wireless LAN's. In Proceedings of the Conference on Communications Architectures, Protocols and Applications, London, UK, 31 August–2 September 1994; pp. 212–225.
45. Tobagi, A.F.; Kleinrock, L. Packet Switching in Radio Channels: Part II—The Hidden Terminal Problem in Carrier Sense Multiple-Access and the Busy-Tone Solution. *IEEE Trans. Commun.* **1975**, *23*, 1417–1433. [CrossRef]
46. Jin, T.K.; Cho, H.D. Multi-Code MAC for Multi-Hop Wireless Ad hoc Networks. *IEEE Vehic. Tech. Conf.* **2002**, *2*, 1100–1104.

47. Borgonovo, F.; Capone, A.; Cesana, M.; Fratta, L. ADHOC MAC: A New MAC Architecture for Ad Hoc Networks Providing Efficient and Reliable Point-to-Point and Broadcast Services. *Wirel. Netw.* **2004**, *10*, 359–366. [CrossRef]
48. Crowther, W.; Rettberg, R.; Walden, D.; Ornstein, S.; Heart, F. A System for Broadcast Communications: Reservation ALOHA. In Proceedings of the 6th Hawaii International Conference on Systems Sciences, Honolulu, HI, USA, 9–11 January 1973; pp. 596–603.
49. Borgonovo, F.; Capone, A.; Cesana, M.; Fratta, L. RR-ALOHA, a Reliable R-ALOHA broadcast channel for ad-hoc inter-vehicle communication networks. In Proceedings of the Med-Hoc-Net, Sardegna, Italy, 4–6 September 2002; Available online: <https://www.researchgate.net/publication/228598697> (accessed on 8 November 2021).
50. Nasipuri, A.; Ye, S.; You, J.; Hiromoto, E.R. A MAC Protocol for Mobile Ad Hoc Networks Using Directional Antennas. In Proceedings of the Wireless Communications and Networking Conference, Chicago, IL, USA, 23–28 September 2000; Volume 1, pp. 1214–1219.
51. 802.11p-2010. In *IEEE Standard for Information Technology Telecommunications and Information Exchange between Systems—Local and Metropolitan Area Networks—Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6*; Wireless Access in Vehicular Environments Std. IEEE: Piscataway, NJ, USA, 2010.
52. Guo, W.; Huang, L.; Chen, L.; Xu, H.; Xie, J. An adaptive collision-free MAC protocol based on TDMA for inter-vehicular communication. In Proceedings of the International Conference on Wireless Communications and Signal Processing (WCSP), Huangshan, China, 25–27 October 2012; pp. 1–6.
53. Omar, A.H.; Zhuang, W. VeMAC: A TDMA-Based MAC Protocol for Reliable Broadcast in VANETs. *IEEE Trans. Mob. Comput.* **2013**, *12*, 1724–1736. [CrossRef]
54. Yu, F.; Biswas, S. Self-configuring TDMA protocols for enhancing vehicle safety with DSRC based vehicle-to-vehicle communications. *IEEE J. Sel. Areas Commun.* **2007**, *25*, 1526–1537. [CrossRef]
55. Bharati, S.; Zhuang, W. CAH-MAC: Cooperative ADHOC MAC for vehicular networks. *IEEE J. Sel. Areas Commun.* **2013**, *31*, 470–479. [CrossRef]
56. Liu, J.; Ren, F.; Miao, L.; Lin, C. A-ADHOC: An adaptive real-time distributed MAC protocol for Vehicular Ad Hoc Networks. *Mob. Netw. Appl.* **2011**, *16*, 576–585. [CrossRef]
57. Abdalla, M.G.; Abu-Rgheff, A.M.; Senouci, M.S. Space-Orthogonal Frequency Time Medium Access Control (SOFT MAC) in VANET. In Proceedings of the IEEE Global Information Infrastructure Symposium, Hammamet, Tunisia, 23–26 June 2009; pp. 1–8.
58. Dang, D.N.M.; Dang, H.N.; Nguyen, V.; Htike, Z.; Hong, C.S. HER-MAC: A hybrid efficient and reliable MAC for vehicular ad hoc networks. In Proceedings of the IEEE 28th International Conference on Advanced Information Networking and Applications, Victoria, BC, Canada, 13–16 May 2014; pp. 186–193.
59. Nguyen, V.; Oo, T.Z.; Chuan, P.; Hong, C.S. An efficient time slot acquisition on the hybrid TDMA/CSMA multichannel MAC in VANETs. *IEEE Commun. Lett.* **2016**, *20*, 970–973. [CrossRef]
60. Lu, N.; Ji, Y.; Liu, F.; Wang, X. A dedicated multi-channel MAC protocol design for VANET with adaptive broadcasting. In Proceedings of the IEEE Wireless Communication and Networking Conference, Sydney, NSW, Australia, 18–21 April 2010; pp. 1–6.
61. Nguyen, V.; Pham, C.; Thant Zin Oo, Z.H.; Tran, H.N.; Huh, N.E.; Hong, S.C. MAC protocols with dynamic interval schemes for VANETs. *Veh. Commun.* **2018**, *15*, 4–62. [CrossRef]
62. Mukhtaruzzaman, M.; Atiquzzaman, M. Clustering in VANET: Algorithms and Challenges. *J. Comput. Electr. Eng.* **2020**, *88*, 106851. Available online: <https://arxiv.org/ftp/arxiv/papers/2009/2009.01964.pdf> (accessed on 8 November 2021). [CrossRef]
63. Chen, W.; Jia, J.; Du, X.; Zhang, Y.; Liu, K. A Clustering-Based Adaptive Multiple Access Protocol for Vehicular Ad Hoc Networks. In Proceedings of the International Conference on Communication Systems, Chengdu, China, 19–21 December 2018; pp. 463–467.
64. Senouci, O.; Harous, S.; Aliouat, Z. Survey on vehicular ad hoc networks clustering algorithms: Overview, taxonomy, challenges, and open research issues. *Int. J. Commun. Syst.* **2020**, *33*, e4402. [CrossRef]
65. Al-heety, S.O.; Zakaria, Z.; Ismail, M.; Shakir, M.M.; Alani, S.; Alsariera, H. A Comprehensive Survey: Benefits, Services, Recent Works, Challenges, Security, and Use Cases for SDN-VANET. *IEEE Access* **2020**, *8*, 91028–91047. [CrossRef]
66. Jaballah, B.W.; Conti, M.; Lal, C. Software-Defined VANETs: Benefits, Challenges, and Future Directions, Networking and Internet Architecture. *arXiv* **2019**, arXiv:1904.04577.
67. Arena, F.; Pau, G.; Severino, A. A review on IEEE 802.11 p for intelligent transportation systems. *J. Sens. Actuator Netw.* **2020**, *9*, 22. [CrossRef]
68. Naik, G.; Choudhury, B.; Park, J.M. IEEE 802.11 bd & 5G NR V2X: Evolution of radio access technologies for V2X communications. *IEEE Access* **2019**, *7*, 70169–70184.
69. Peng, S.; Zhou, L.; He, X.; Du, J. GPS-aided inter-microcell interference avoidance for request-transmission splitting slotted ALOHA-based scheme in smart cities with connected vehicles. *Future Gener. Comput. Syst.* **2020**, *108*, 501–511. [CrossRef]
70. Jiang, X.; Du, D.H.C. PTMAC: A Prediction-based TDMA MAC Protocol for Reducing Packet Collisions in VANET. *IEEE Trans. Veh. Technol.* **2016**, *65*, 9209–9223. [CrossRef]
71. Chen, P.; Zheng, J.; Wu, Y. A-VeMAC: An Adaptive Vehicular MAC Protocol for Vehicular Ad Hoc Networks. In Proceedings of the ICC—Ad-Hoc and Sensor Networks Symposium, Paris, France, 21–25 May 2017; pp. 1–6.
72. Ndihi, N.D.E.; Cherkaoui, S. Modulation Network Coding. *IET Commun.* **2013**, *7*, 1715–1728. [CrossRef]
73. Ndihi, N.D.E.; Cherkaoui, S. MAP: Contention-Free MAC protocol for VANETs with PLNC. In Proceedings of the ICC-2017 Mobile and Wireless Networking, Paris, France, 21–25 May 2017; pp. 1–7.

74. Urmonov, O.; Kim, H. Highly Reliable MAC Protocol Based on Associative Acknowledgement for Vehicular Network. *Electronics* **2021**, *10*, 382. [[CrossRef](#)]
75. Mao, Q.; Yue, P.; Xu, M.; Ji, Y.; Cui, Z. OCTMAC: A VLC Based MAC Protocol Combining Optical CDMA with TDMA for VANETs. In Proceedings of the International Conference on Computer, Information and Telecommunication Systems (CITS), Dalian, China, 21–23 July 2017; pp. 234–238.
76. Guerra-Medina, F.M.; Rojas-Guillama, B.; Gonzalez, M.A.J.; Poves, E. Experimental optical code-division multiple access system for visible light communications. In Proceedings of the 2011 Wireless Telecommunications Symposium (WTS), New York, NY, USA, 13–15 April 2011; pp. 1–6.
77. Cao, Y.; Zhang, H.; Fang, Y.; Yuan, D. An Adaptive High-Throughput Multi-Channel MAC Protocol for VANETs. *IEEE Internet Things J.* **2020**, *7*, 8249–8262. [[CrossRef](#)]
78. Li, S.; Liu, Y.; Wang, J.; Ge, Y.; Deng, L.; Deng, W. TCGMAC: A TDMA-based MAC protocol with collision alleviation based on slot declaration and game theory in VANETS. *Trans. Emerg. Telecommun. Technol.* **2019**, *30*, e3730. [[CrossRef](#)]
79. Yanez, A.; Cespedes, S.; Rubio-Loyola, J. CaSSaM: Context-aware System for Safety Messages Dissemination in VANETS. In Proceedings of the IEEE COLCOM, Medellin, Colombia, 16–18 May 2018; pp. 1–6. ISBN 978-1-5386-6820-7/18.
80. Sutton, S.R.; Barto, G.A. *Introduction to Reinforcement Learning*, 1st ed.; MIT Press: Cambridge, MA, USA, 1998.
81. Watkins, H.C.J.C.; Dayan, P. Q-learning. *Mach. Learn.* **1992**, *8*, 279–292. [[CrossRef](#)]
82. Lyu, F.; Zhu, H.; Zhou, H.; Qian, L.; Xu, W.; Li, M.; Shen, S.X. MoMAC: Mobility-Aware and Collision Avoidance MAC for Safety Applications in VANETS. *IEEE Trans. Veh. Technol.* **2018**, *67*, 10590–10602. [[CrossRef](#)]
83. Samara, G. An Efficient Collision Free Protocol for VANET. *Int. J. Comput. Appl.* **2018**, *180*, 30–34. [[CrossRef](#)]
84. Lyu, F.; Zhu, H.; Zhou, H.; Xu, W.; Zhang, N.; Li, M.; Shen, S.X. SS-MAC: A Novel Time Slot-Sharing MAC for Safety Messages Broadcasting in VANETS. *IEEE Trans. Veh. Technol.* **2018**, *67*, 3586–3597. [[CrossRef](#)]
85. Cao, S.; Lee, S.C.V. A Novel Adaptive TDMA-Based MAC Protocol for VANETS. *IEEE Commun. Lett.* **2018**, *22*, 614–617. [[CrossRef](#)]
86. Siddik, M.A.; Moni, S.S.; Alam, M.S.; Johnson, W.A. SAFE-MAC: Speed Aware Fairness Enabled MAC Protocol for Vehicular Ad-hoc Networks. *Sensors* **2019**, *19*, 2405. [[CrossRef](#)]
87. AĞÜN, F.; Çibuk, M.; UR-Rehman, S. ReMAC: A novel hybrid and reservation-based MAC protocol for VANETS. *Turk. J. Electr. Eng. Comput. Sci.* **2020**, *28*, 1886–1904. [[CrossRef](#)]
88. Boulila, N.; Haddad, M.; Laouiti, A.; Saidane, A.L. QCH-MAC: A Qosaware Centralized Hybrid MAC protocol for Vehicular Ad Hoc NETWORKS. In Proceedings of the 32nd International Conference on Advanced Information Networking and Applications (AINA), Krakow, Poland, 16–18 May 2018; pp. 55–62. [[CrossRef](#)]
89. Karabulut, A.M.; Shahen Shah, M.F.A.; İlhan, H. CR-MAC: Cooperative Relaying MAC Protocol for VANETS. In Proceedings of the Scientific Meeting on Electrical-Electronics and Biomedical Engineering and Computer Science (EBBT), Istanbul, Turkey, 24–26 April 2019; pp. 1–4. ISBN 978-1-7281-1013-4/19.
90. Karabulut, A.M.; Shahen Shah, M.F.A.; İlhan, H. CoMACAV: Cooperative MAC Protocol for Autonomous Vehicles. In Proceedings of the IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA), Sofia, Bulgaria, 3–5 July 2019; pp. 1–8. ISBN 978-1-7281-1862-8/19.
91. Lu, X.; Xu, D.; Xiao, L.; Wang, L.; Zhuang, W. Anti-jamming communication game for UAV-aided VANETS. In Proceedings of the IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6.
92. Benslimane, A.; Nguyen-Minh, H. Jamming attack model and detection method for beacons under multi-channel operation in vehicular networks. *IEEE Trans. Veh. Technol.* **2016**, *66*, 6475–6488. [[CrossRef](#)]
93. Cormen, H.T.; Leiserson, E.C.; Rivest, R.L.; Stein, C. *Introduction to Algorithms*, 3rd ed.; MIT Press: Cambridge, MA, USA, 2009.
94. Jeong, J.; Shen, Y.; Sangsoo Jeong, S.; Lee, S.; Jeong, H.; Oh, J.; Park, T.; Ilyas, U.M.; Son, H.S.; Du, H.C.D. STMAC: Spatio-Temporal Coordination-Based MAC Protocol for Driving Safety in Urban Vehicular Networks. *IEEE Trans. Intell. Transp.* **2017**, *19*, 1520–1536. [[CrossRef](#)]
95. Abbas, G.; Abbas, H.Z.; Haider, S.; Baker, T.; Boudjit, S.; Muhammad, F. PDMAC: A Priority-Based Enhanced TDMA Protocol for Warning Message Dissemination in VANETS. *Sensors* **2020**, *20*, 45. [[CrossRef](#)]
96. Shahen Shah, M.F.A.; İlhan, H.; Tureli, U. CB-MAC: A novel cluster-based MAC protocol for VANETS. *IET Intell. Transp. Syst.* **2019**, *13*, 587–595. [[CrossRef](#)]
97. Daknou, E.; Tabbane, N.; Thaalbi, M. A MAC Multi-channel Scheme based on Learning Automata for Clustered VANETS. In Proceedings of the 32nd International Conference on Advanced Information Networking and Applications, Cracow, Poland, 16–18 May 2018; pp. 71–78.
98. Aredla, J.; Zhang, S.; Liu, E. Relia-MAC: Timely and Reliable Communication for Safety Critical Applications of VANET. In Proceedings of the 25th International Conference on Automation and Computing, Lancaster University, Lancaster, UK, 5–7 September 2019; pp. 1–6.
99. Roman, B.; Wassell, I.; Chatzigeorgiou, I. Scalable Cross-Layer Wireless Access Control Using Multi-Carrier Burst Contention. *IEEE J. Sel. Areas Commun.* **2011**, *29*, 113–128. [[CrossRef](#)]
100. Cao, Y.; Zhang, H.; Wu, D.; Yuan, D. OGCMAC: A Novel OFDM based Group Contention MAC for VANET Control Channel. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 5796–5809. [[CrossRef](#)]
101. Shen, Z.; Zhang, X.; Zhang, M.; Li, W.; Yang, D. Self-Sorting Based MAC Protocol for High-Density Vehicular Ad Hoc Networks. *IEEE Access* **2017**, *5*, 7350–7361. [[CrossRef](#)]

102. Peng, S.; Zhou, L.; He, X.; Du, J.; Guan, L.Y. A request-transmission splitting scheme for V2X networks: Design and throughput analysis. *IEEE Access* **2019**, *7*, 158317–158325. [\[CrossRef\]](#)
103. Zhuo, X.; Hua, S.; Miao, L.; Dai, Y. Direction matters: A decentralized direction-based tdma scheduling strategy for vanet. In Proceedings of the 13th International Conference on Communication Technology (ICCT), Jinan, China, 25–28 September 2011; pp. 566–571.
104. Samara, G. An improved CF-MAC protocol for VANET. *Int. J. Electr. Comput. Eng.* **2019**, *9*, 2668–2674. [\[CrossRef\]](#)
105. Balador, A.; Calafate, T.C.; Cano, C.J.; Manzoni, P. DTB-MAC: Dynamic Token-Based MAC Protocol for Reliable and Efficient Beacon Broadcasting in VANETs. In Proceedings of the 12th Annual IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, USA, 9–12 January 2015; pp. 109–114.
106. Aggun, F.; Cibuk, M.; Rehman, U.S. A new Self-Organizing Multi-channel MAC schema for RSU-centric VANETs. *Phys. A Stat. Mech. Appl.* **2020**, *551*, 124098. [\[CrossRef\]](#)
107. Ndihi, N.D.E.; Cherkaoui, S. Mac for Physical-Layer Network Coding in VANETs. *Int. J. Bus. Data Commun.* **2012**, *8*, 84–106. [\[CrossRef\]](#)
108. El-Gawad, M.A.A.; ElSharief, M.; Kim, H. A cooperative V2X MAC protocol for vehicular networks. *EURASIP J. Wirel. Commun. Netw.* **2019**, *2019*, 65. [\[CrossRef\]](#)
109. Zheng, Z.; Sangaiah, K.; Wang, T. Adaptive Communication Protocols in Flying Ad Hoc Network. *IEEE Commun. Mag.* **2018**, *56*, 136–142. [\[CrossRef\]](#)
110. Li, S.; Liu, Y.; Wang, J.; Sun, Z. SCMAC: A Slotted-Contention-Based Media Access Control Protocol for Cooperative Safety in VANETs. *IEEE Internet Things J.* **2020**, *7*, 3812–3821. [\[CrossRef\]](#)
111. Zhang, X.; Jiang, X.; Zhang, M. A black-burst based time slot acquisition scheme for the hybrid TDMA/CSMA multichannel MAC in VANETs. *IEEE Wirel. Commun. Lett.* **2019**, *8*, 137–140. [\[CrossRef\]](#)
112. Lyu, F.; Cheng, N.; Zhou, H.; Xu, W.; Shi, W.; Chen, J.; Li, M. DBCC: Leveraging link perception for distributed beacon congestion control in VANETs. *IEEE Internet Things J.* **2018**, *5*, 4237–4249. [\[CrossRef\]](#)
113. Verma, P.; Singh, N.; Sharma, M. Modelling a vehicle-ID-based IEEE 802.11 OCB MAC scheme for periodic broadcast in vehicular networks. *IET Commun.* **2018**, *12*, 2401–2407. [\[CrossRef\]](#)
114. Pressas, A.; Sheng, Z.; Ali, F.; Tian, D.; Nekovee, M. Contention-based Learning MAC Protocol for Broadcast Vehicle-to-Vehicle Communication. In Proceedings of the IEEE Vehicular Networking Conference (VNC), Taipei, Taiwan, 5–7 December 2018; pp. 263–270. [\[CrossRef\]](#)
115. Qiu, F.J.H.; Ho, W.I.; Tse, K.C.; Xie, Y. Technical Report: A Methodology for Studying 802.11p VANET Broadcasting Performance with Practical Vehicle Distribution. *IEEE Trans. Veh. Technol.* **2015**, *64*, 1939–1959. [\[CrossRef\]](#)
116. Lyu, F.; Zhu, H.; Cheng, N.; Zhu, Y.; Zhou, H.; Xu, W.; Xue, G.; Li, M. ABC: Adaptive Beacon Control for Rear-End Collision Avoidance in VANETs. In Proceedings of the 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), Hong Kong, China, 11–13 June 2018; pp. 1–9. ISBN 978-1-5386-4281-8/18.
117. Kenney, B.J.; Bansal, G.; Rohrs, E.C. LIMERIC: A Linear Message Rate Control Algorithm for Vehicular DSRC Systems. *IEEE Trans. Veh. Technol.* **2013**, *62*, 4182–4197. [\[CrossRef\]](#)
118. Yang, W.; Liu, W.; Li, P.; Sun, L. TDMA-Based Control Channel Access for IEEE 802.11p in VANETs. *Int. J. Distrib. Sens. Netw.* **2014**, *2014*, 579791. [\[CrossRef\]](#)
119. Haddad, M.; Laouiti, A.; Zagrouba, R.; Muhlethaler, P.; Saidane, A.L. *A Fully Distributed TDMA based MAC Protocol for Vehicular Ad Hoc Networks*; Research Report; Inria Paris Rocquen Court. HAL-Inria: Lyon, France, 2015. Available online: <https://hal.archives-ouvertes.fr/hal-01211456/document> (accessed on 8 November 2021).
120. Karamad, E.; Ashtiani, F. Performance analysis of IEEE 802.11 DCF and 802.11e EDCA based on queueing networks. *IET Commun.* **2009**, *3*, 871–881. [\[CrossRef\]](#)
121. Chung, M.J.; Kim, M.; Park, S.Y.; Choi, M.; Lee, S.; Oh, S.H. Time coordinated V2I communications and handover for WAVE networks. *IEEE J. Sel. Areas Commun.* **2011**, *29*, 545–558. [\[CrossRef\]](#)
122. Cheng, S.T.; Horng, G.J.; Chou, C.L. Using Cellular Automata to Form Car Society in Vehicular Ad Hoc Networks. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 1374–1384. [\[CrossRef\]](#)
123. Zafar, A.B.; Boulila, N.; Saidane, L. Communication delay guarantee for IEEE 802.11p/wave Vehicle networks with RSU control. In Proceedings of the 13th International Conference on Networking, Sensing, and Control (ICNSC), Mexico City, Mexico, 28–30 April 2016; pp. 1–7. [\[CrossRef\]](#)
124. Zhang, Y.; Ye, M.; Guan, L. Overlap-Minimization Scheduling Strategy for Data Transmission in VANET. In Proceedings of the 2021 International Symposium on Networks, Computers and Communications (ISNCC), Dubai, United Arab Emirates, 31 October–2 November 2021.
125. Kim, S.; Stark, E.W. Simple relay enabled MAC (SRMAC) protocol for cooperative communication. In Proceedings of the IEEE Military Communications Conference, San Diego, CA, USA, 18–20 November 2013; pp. 175–180.
126. Wang, S.; Liu, J.; Zhou, S.; Zhou, L.; Yin, M.; Hao, H. Cooperative relay MAC protocol for ad hoc networks. In Proceedings of the 17th International Conference on Communication Technology (ICCT), Chengdu, China, 27–30 October 2017; pp. 612–616.
127. Xiao, L.; Lu, X.; Xu, D.; Tang, Y.; Wang, L.; Zhuang, W. UAV Relay in VANETs Against Smart Jamming with Reinforcement Learning. *IEEE Trans. Veh. Technol.* **2018**, *67*, 4087–4097. [\[CrossRef\]](#)

128. Valentina, M.G.; Jayashri, S. Q-Learning based point to point data transfer in Vanets. In Proceedings of the 3rd International Conference on Recent Trends in Computing (ICRTC), Ghaziabad, India, 12–13 March 2015; Volume 57, pp. 1394–1400.
129. Karabulut, A.M.; Shahen Shah, M.F.A.; Ilhan, H. OEC-MAC: A Novel OFDMA Based Efficient Cooperative MAC Protocol for VANETS. *IEEE Access* **2020**, *8*, 94665–94677. [[CrossRef](#)]
130. Shahen Shah, M.F.A.; Ilhan, H.; Tureli, U. RECV-MAC: A novel reliable and efficient cooperative MAC protocol for VANETS. *IET Commun.* **2019**, *13*, 2541–2549. [[CrossRef](#)]
131. Liu, Y.; Zhou, H.; Huang, J. OCA-MAC: A Cooperative TDMA-Based MAC Protocol for Vehicular Ad Hoc Networks. *Sensors* **2019**, *19*, 2691. [[CrossRef](#)] [[PubMed](#)]
132. Yoon, H.Y.; Ko, B.Y. CTMAC: A Cooperative TDMA MAC in Vehicular Ad Hoc Networks. In Proceedings of the International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 18–20 October 2017; pp. 772–774. [[CrossRef](#)]
133. Feng, T.K. LMA: Location- and Mobility-Aware Medium-Access Control Protocols for Vehicular Ad Hoc Networks Using Directional Antennas. *IEEE Trans. Veh. Technol.* **2007**, *56*, 3324–3336. [[CrossRef](#)]
134. Huang, W.; Mao, Y.; Min, X.; Hu, J.; Shen, L. Analysis of SFR's Impaction on Delay Time Based DMMAC for VANETS. In Proceedings of the 7th International Conference on Wireless Communications & Signal Processing (WCSP), Nanjing, China, 15–17 October 2015; pp. 1–5. [[CrossRef](#)]
135. Lin, Z.; Tang, Y. Distributed multi-channel MAC protocol for VANET: An adaptive frame structure scheme. *IEEE Access* **2019**, *7*, 12868–12878. [[CrossRef](#)]
136. Salem, F.; Elhillali, Y.; Nisar, S. Efficient modelling of IEEE 802.11p MAC output process for V2X interworking enhancement. *IET Netw.* **2018**, *7*, 210–219. [[CrossRef](#)]
137. Khanafer, M.; Kandil, M.; Al-Baghdadi, R.; Al-Ajmi, A.; Mouftah, T.H. Enhancements to IEEE 802.15.4 MAC Protocol to Support Vehicle-to-Roadside Communications in VANETS. In Proceedings of the IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019; p. 15. [[CrossRef](#)]
138. Khanafer, M.; Guennoun, M.; Mouftah, H. A Survey of Beacon Enabled IEEE 802.15.4 MAC Protocols in Wireless Sensor Networks. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 856–876. [[CrossRef](#)]
139. Almalag, S.M.; Olariu, S.; Weigle, C.M. TDMA Cluster-based MAC for VANETS (TC-MAC). In Proceedings of the International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), San Francisco, CA, USA, 25–28 June 2012; pp. 1–6. [[CrossRef](#)]
140. Hadded, M.; Zagrouba, R.; Laouiti, A.; Muhlethaler, P.; Saïdane, A.L. An Adaptive TDMA Slot Assignment Strategy in Vehicular Ad Hoc Networks. *J. Mach. Mach. Commun.* **2014**, *1*, 175–194. [[CrossRef](#)]
141. Shahin, N.; Kim, T.Y. An Enhanced TDMA Cluster-based MAC (ETCM) for Multichannel Vehicular Networks. In Proceedings of the International Conference on Selected Topics in Mobile and Wireless Networking (MoWNeT), Cairo, Egypt, 11–13 April 2016; pp. 1–8. [[CrossRef](#)]
142. Akhter, S.M.F.A.; Ahmed, M.; Shahen Shah, M.F.A.; Anwar, A.; Zengin, A. A Secured Privacy-Preserving Multi-Level Blockchain Framework for Cluster Based VANET. *Sustainability* **2021**, *13*, 400. [[CrossRef](#)]
143. Li, M.; Lou, W.; Zeng, K. OppCast: Opportunistic Broadcast of Warning Messages in VANETS with Unreliable Links. In Proceedings of the 6th International Conference on Mobile Adhoc and Sensor Systems, Macau, China, 12–15 October 2009; pp. 534–543. [[CrossRef](#)]





## Article

# Online Service Function Chain Deployment for Live-Streaming in Virtualized Content Delivery Networks: A Deep Reinforcement Learning Approach

Jesús Fernando Cevallos Moreno <sup>1,†</sup>, Rebecca Sattler <sup>2,†</sup>, Raúl P. Caulier Cisterna <sup>3,†</sup>, Lorenzo Ricciardi Celsi <sup>4</sup>, Amina Sánchez Rodríguez <sup>5,\*</sup> and Massimo Mecella <sup>1,†</sup>

<sup>1</sup> Department of Computer Science, Automation and Management Engineering, Sapienza University of Rome, Via Ariosto 25, 00185 Rome, Italy; cevallos@diag.uniroma1.it (J.F.C.M.); mecella@diag.uniroma1.it (M.M.)

<sup>2</sup> Department of Computer Science, Databases and Information Systems, Humboldt University of Berlin, Unter den Linden 6, 10099 Berlin, Germany; rebecca.sattler@hu-berlin.de

<sup>3</sup> Centro de Imagen Biomédica, Pontificia Universidad Católica de Chile, Av. Vicuña Mackenna 4860, Macul 7820436, Chile; raul.caulier@uc.cl

<sup>4</sup> ELIS Innovation Hub, Via Sandro Sandri 45-81, 00159 Rome, Italy; lricciardicelsi@elis.org

<sup>5</sup> Microbial Systems Ecology and Evolution Hub, Universidad Técnica Particular de Loja, Loja 1101608, Ecuador

\* Correspondence: asanchez2@utpl.edu.ec

† These authors are also with The Riello Institute for Integral Development, 25-40 Shore Blvd, PH 20R, Astoria, NY 11102, USA.

**Citation:** Cevallos Moreno, J.F.; Sattler, R.; Caulier Cisterna, R.P.; Ricciardi Celsi, L.; Sánchez Rodríguez, A.; Mecella, M. Online Service Function Chain Deployment for Live-Streaming in Virtualized Content Delivery Networks: A Deep Reinforcement Learning Approach. *Future Internet* **2021**, *13*, 278. <https://doi.org/10.3390/fi13110278>

Academic Editor: Michael Mackay

Received: 8 October 2021

Accepted: 27 October 2021

Published: 29 October 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Abstract:** Video delivery is exploiting 5G networks to enable higher server consolidation and deployment flexibility. Performance optimization is also a key target in such network systems. We present a multi-objective optimization framework for service function chain deployment in the particular context of Live-Streaming in virtualized content delivery networks using deep reinforcement learning. We use an Enhanced Exploration, Dense-reward mechanism over a Dueling Double Deep Q Network (E2-D4QN). Our model assumes to use network function virtualization at the container level. We carefully model processing times as a function of current resource utilization in data ingestion and streaming processes. We assess the performance of our algorithm under bounded network resource conditions to build a safe exploration strategy that enables the market entry of new bounded-budget vCDN players. Trace-driven simulations with real-world data reveal that our approach is the only one to adapt to the complexity of the particular context of Live-Video delivery concerning the state-of-art algorithms designed for general-case service function chain deployment. In particular, our simulation test revealed a substantial QoS/QoE performance improvement in terms of session acceptance ratio against the compared algorithms while keeping operational costs within proper bounds.

**Keywords:** live-video delivery; 5G networks; virtualized content delivery networks; network function virtualization; service function chain deployment; deep reinforcement learning

## 1. Introduction

Video traffic occupies more than three-quarters of total internet traffic nowadays, and the trend is to grow [1]. Such growth is mainly characterized by Over-The-Top (OTT) video delivery. Content providers need OTT Content Delivery systems to be efficient, scalable, and adaptive [2]. For this reason, cost and Quality of Service (QoS) optimization for video content delivery systems is an active research area. A lot of the research effort in this context is being placed on the optimization [3], and modeling [4] of Content Delivery systems' performance.

Content Delivery Networks (CDN) are distributed systems that optimize the end-to-end delay of content requests over a network. CDN systems are based on the redirection of requests and content reflection. Well-designed CDN systems warrant good Quality of

service (QoS) and Quality of Experience (QoE) for widely distributed users. CDN uses replica servers as proxies for the content providers' origin servers. Traditional CDN systems are deployed on dedicated hardware and incur, thereby in high capital and operational expenditures. On the other hand, virtualized Content Delivery Networks (vCDN) use Network Function Virtualization (NFV) [5], and Software-defined Networking (SDN) [6] to deploy their software components on virtualized network infrastructures like virtual machines or containers. NFV and SDN are key enablers for 5G networks [7]. Over the last two years, numerous internet services providers (ISP), mobile virtual network operators (MVNO), and other network market players are exploiting 5G networks to augment the spectrum of network services they offer [8]. In this context, virtualized CDN systems are taking the profit of 5G networks to enable higher server consolidation, and flexibility during deployment [9,10]. vCDNs reduce both capital and operational expenditures concerning CDNs deployed to dedicated-hardware [11]. Further, vCDNs are edge-computing compliant [12] and make possible to act win-win strategies between ISP and CDN providers [13].

### 1.1. Problem Definition

Virtualized Network systems are usually deployed as a composite chain of Virtual Network Functions (VNF), often called a *service function chain* (SFC). Every incoming request to a virtualized network system will be mapped to a corresponding deployed SFC. The problem of deploying a SFC inside a VNF infrastructure is called *VNF Placement* or *SFC Deployment* [14]. Many service requests can share the same SFC deployment scheme, or the SFC deployments can vary. Given two service requests that share the same requested chain of VNFs, the SFC deployment will vary when at least one pair of same-type VNFs are deployed on different physical locations for each request.

This work focuses on the particular case of Live-Video delivery, also referred to as *live-streaming*. In such a context, each service request is associated with a Live-Video streaming session. CDNs have proved essential to meet scalability, reliability, and security in Live-Video delivery scenarios. One important Quality of Experience (QoE) measure in live-video streaming is the *session startup delay*, which is the time the end-user waits since the content is requested and the video is displayed. One important factor that influences the startup delay is the round-trip-time (RTT) of the session request, which is the time between the content request is sent, and the response is received. In live-Streaming, the data requested by each session is determined only by the particular content provider or *channel* requested. Notably, cache HIT and cache MISS events may result in very different request RTTs. Consequently, a realistic Live-Streaming vCDN model should keep track of the caching memory status of every cache-VNF module for fine-grain RTT simulation.

Different SFC deployments may result in different round-trip times (RTT) for live-video sessions. The QoS/QoE goodness of a particular SFC deployment policy is generally measured by the mean acceptance ratio (AR) of client requests, where the acceptance ratio is defined as the percentage of requests whose RTT is below a maximum threshold [14–16]. Notice that RTT is different from the total delay, which is the total propagation time of the data stream from the origin server and the end-user.

Another important factor that influences RTT computation is the request processing time. Such a processing time will notably depend on the current VNF utilization. To model VNF utilization in a video-delivery context, major video streaming companies [17] recommend to consider not only the *content-delivery* tasks, but also the resource consumption associated with *content-ingestion* processes. In other words, any VNF must ingest a particular data stream before being able to deliver it through its own client connections, and such ingestion will incur non-negligible resource usage. Further, a realistic vCDN delay model must incorporate VNF instantiation times, as they may notably augment the starting delay of any video-streaming session. Finally, both instantiation time and resource consumption may differ significantly depending on the specific characteristics of each VNF [3].

In this paper, we model a vCDN following the NFV Management and orchestration (NFV-MANO) framework published by the ETSI standard group specification [9,18–20]. We propose elastic container-based virtualization of a CDN inspired by [5,19]. One of the management software components of a vCDN in the ETSI standard is the Virtual Network Orchestrator (VNO), which is generally responsible for the dynamic scaling of the containers' resource provision [19]. The resource scaling is triggered when needed, for example, when traffic bursts occur. Such a resource scaling may influence the resource provision costs, also called *hosting costs*, especially if we consider a cloud-hosted vCDN context [20]. Also, data-transportation (DT) routes in the substrate network of a vCDN may be affected by different SFC deployment decisions. Consequently, DT costs may also vary [9], especially if we consider a multi-cloud environment as we do in this work. Thus, DT costs may also be an important part of the operational costs of a vCDN.

Finally, *Online* SFC Deployment implies taking fine-grained control decisions over the deployment scheme of SFCs when the system is running. For example, one could associate one SFC deployment for each incoming request to the system or keep the same SFC deployment for different requests but be able to change it whenever requested. In all cases, Online SFC Deployment implies that SFC policy adaptations can be done each time a new SFC request comes into the system. *Offline* deployment instead takes one-time decisions over aggregations of input traffic to the system [21]. Offline optimization's effectiveness relies on the estimation accuracy of future environment characteristics [22]. Such estimation may not be trivial, especially in contexts where incoming traffic patterns may be unpredictable or when request characteristics might be heterogeneous, which is the case of Live-Streaming [9,14,23,24]. Consequently, in this work, we consider the *Online* optimization of a Live-Streaming vCDN SFC deployment, and we seek such optimization in terms of both QoS/QoE and operational costs, where the latter are composed of hosting costs and data-transportation costs.

### 1.2. Related Works

SFC Deployment is an NP-hard problem [25]. Several exact optimization models, and sub-optimal heuristics, have been proposed in recent years. Ibn-Khedher et al. [26] defined a protocol for optimal VNF placement based on SDN traffic rules and an exact optimization algorithm. This work solves the optimal VNF placement, migration, and routing problem, modeling various system and network metrics as costs and user satisfaction. HPAC [27] is a heuristic proposed by the same authors to scale the solutions of [26] for bigger topologies based on the Gomory-Hu tree transformation of the network. Their call for future work includes the need for the dynamic triggering of adaptation like monitoring user demands, network loads, and other system parameters. Yala et al. [28] present a resource allocation and VNF placement optimization model for vCDN and a two-phase offline heuristic for solving it in polynomial time. This work models server availability through an empirical probabilistic model and optimizes this score alongside the VNF deployment costs. Their algorithm produces near-optimal solutions in minutes for a Network Function Virtualization Infrastructure (NFVI) deployed on a substrate network made of even 600 physical machines. They base the dimensioning criteria on extensive video streaming VNF QoE-aware benchmarking.

Authors in [20] keep track of resource utilization in the context of an optimization model for multi-cloud placement of VNF chains. Utilization statistics per node and network statistics per link are taken into account inside a simulation/optimization framework for VNF placement in vCDN in [29]. This offline algorithm can handle large-scale graph topologies being designed to run on a parallel-supercomputer environment. This work analyzes the effect of routing strategies on the results of the placement algorithm and performs better with a greedy max-bandwidth routing approach. The caching state of each cache-VNF is modeled with a probabilistic function in this work. Offline Optimization of Value Added Service (VAS) Chains in vCDN is proposed in [30], where authors model an Integer Linear Programming (ILP) problem to optimize QoS and Provider Costs. This work

models license costs for each VNF added in a new physical location. An online alternative is presented in [31], where authors model the cost of VNF instantiations when optimizing online VNF placement for vCDN. This model lacks to penalize the Round Trip Time (RTT) of requests with the instantiation time of such VNFs. More scalable solutions for this problem are leveraged with heuristic-based approaches like the one in [32]. On the other hand, regularization-based techniques are used to present an online VNF placement algorithm for geo-distributed VNF chain requests in [32]. This work optimizes different costs and the end-to-end delay providing near-optimal solutions in polynomial time.

Robust Optimization (RO) has also been applied to solve various network-related optimization problems. RO and stochastic programming techniques have been used to model optimization under scenarios characterized by data uncertainty. Uncertainty concerning network traffic fluctuations or resource request amount can be modeled if one seeks to minimize network power consumption [33] or the costs related to cloud resource provisioning [34], for example. In [35], Marotta et al. present a VNF placement and SFC routing optimization model that minimizes power consumption taking into account that resource requirements are uncertain and fluctuate in virtual Evolved Packet Core scenarios. Such an algorithm is enhanced in a successive work [36] where authors improve the scalability of their solution by dividing the task into sub-problems and adopting various heuristics. Such an improvement permits solving high-scale VNF placement in less than a second, making such an algorithm suitable for online optimization. Remarkably, the congestion-induced delay has been modeled in this work. Ito et al. [37] instead provide various models of the VNF placement problem where the objective is to warrant probabilistic failure recovery with minimum backup required capacity. Authors in [37] model uncertainty in both failure events and virtual machine capacity.

Deep Reinforcement Learning (DRL) based approaches have been modeled to solve the SFC deployment also. DRL algorithms have recently evolved to solve problems on high-dimensional action spaces through the usage of state-space discretization [38], Policy Learning [39,40], and sophisticated Value learning algorithms [41]. Network-related problems like routing [42], and VNF forwarding graph embedding [43–45] have been solved with DRL techniques. Authors in [46] use the Deep Q-learning framework to implement a VNF placement algorithm which is aware of the server reliability. A policy learning algorithm is used for optimizing operational costs and network throughput on SFC Deployment optimization in [14]. A fault-tolerant version of SFC Deployment is presented in [47], where authors use a Double Deep Q-network (DDQN) and propose different resource reservation schemes to balance the waste of resources and ensure service reliability. Authors in [48] assume to have accurate incoming traffic predictions in input and use a DDQN based algorithm for choosing small-scale network sub-regions to optimize every 15 min. Such a work uses a threshold-based policy to optimize the number of fixed-dimensioned VNF instances. A Proximal Policy Optimization DRL scheme is used in [49] to jointly minimize packet loss and server energy consumption on a cellular network SFC deployment environment. The advantage of DRL approaches with respect to traditional optimization models is the constant time complexity reached after training. A well-designed DRL framework has the potential to achieve complex feature learning and near-optimal solutions even to unprecedented context situations [23].

### 1.3. Main Contribution

To the best of the authors' knowledge, this work is the first VNF-SCF deployment optimization model for the particular case of Live-Streaming vCDN. We propose a vCDN model where we take into account, at the same time:

- VNF-instantiation times,
- Content-delivery and content-ingestion resource usage,
- Utilization-dependant processing times,
- Fine-grained cache-status tracking,
- Operational costs composed of data-transportation costs and hosting costs,

- Multi-cloud deployment characteristics,
- No *a-priory* knowledge of session duration.

We seek to jointly optimize QoS and operational costs with an Online DRL-based approach. To achieve this objective, we propose a *dense-reward* model and an *enhanced-exploration* mechanism for over a dueling-DDQN agent, which combination leads to the convergence to sub-optimal SFC deployment policies.

Further, in this work, we model bounded network resource availability to simulate network overload scenarios. Our aim is to create and validate a *safe-exploration* framework that facilitates the assessment of market-entry conditions for new cloud-hosted Live-Streaming vCDN operators.

Our experiments show that our proposed algorithm is the only one to adapt to the model conditions, maintaining an acceptance ratio above the state-of-art techniques for SFC deployment optimization while keeping a satisfactory balance between network throughput and operational costs. This paper can be seen as an upgrade proposal for the framework presented in [14]. The optimization objective of SFC deployment that we pursue is the same: maximizing the QoS and minimizing the operational costs. We enhance the algorithm used in [14] to find a suitable DRL technique for the particular case of Live-Streaming in v-CDN scenarios.

## 2. Materials and Methods

### 2.1. Problem Modelisation

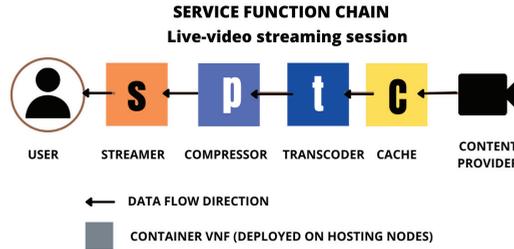
We now rigorously model our SFC Deployment optimization problem. First of all, the system elements that are part of the problem are identified. We then formulate a high-level optimization statement briefly. Successively, our optimization problem's decision variables, penalty terms, and feasibility constraints are described. Finally, we formally define the optimization objective.

#### 2.1.1. Network Elements and Parameters

We model three-node categories in the network infrastructure of a vCDN. The *content provider (CP) nodes*, denoted as  $N_{CP}$ , produce live-video streams that are routed through the SFC to reach the end-users. The VNF *hosting nodes*,  $N_H$ , are the cloud-hosted virtual machines that instantiate container VNFs and interconnect through each other to form the SFCs. Lastly, we consider nodes representing geographic clusters of clients,  $N_{UC}$ . Geographic client clusters are created in such a way that every client in the same geographic cluster is considered to have the same data-propagation delays with respect to the hosting nodes in  $N_H$ . Client cluster nodes will be referred to as *client nodes* from now on. Notice that different hosting nodes may be deployed on different cloud providers. We denote the set of all nodes of the vCDN substrate network as:

$$N = N_{CP} \cup N_H \cup N_{UC}$$

We assume that each live-streaming session request  $r$  is always mapped to a VNF chain containing a Streamer, a Compressor, a Transcoder, and a Cache module [19,50]. In a live-streaming vCDN context, the caching module acts as a proxy that ingests video chunks from a Content Provider, stores them on memory, and sends them to the clients towards the rest of the SFC modules. Caching modules accelerate session startup time and prevent origin server overloads, keep an acceptable total delay, improving session startup times which is a measure of QoE in the context of live-streaming. Compressors, instead, may help to decrease video quality when requested. On the other hand, transcoding functionalities are necessary whenever the requested video codec is different from the original one. Finally, the streamer acts as a multiplexer for the end-users [19]. The order in which the VNF chain is composed is explained by Figure 1.



**Figure 1.** The assumed Service Function Chain composition for every Live-Video Streaming session request. We assume that every incoming session needs for a streamer, a compressor a transcoder and a cache VNF modules. We assume container based virtualization of a vCDN.

We will denote the set of VNF types considered in our model as  $K$ :

$$K = \{streamer, compressor, transcoder, cache\}$$

Any  $k$ -type VNF instantiated at a hosting node  $i$  will be denoted as  $f_i^k, \forall k \in K, \forall i \in N_H$ . We assume that every hosting node is able to instantiate a maximum of one  $k$ -type VNF. Note that, at any time, there might be multiple SFCs whose  $k$ -type module is assigned to a single hosting node  $i$ .

We define fixed-length time windows denoted as  $t, \forall t \in \mathbb{N}$  which we call simulation time-steps following [14]. At each  $t$ , the VNO releases resources for timed-out sessions and processes the incoming session requests denoted as  $R_t = \{r_1, r_2, \dots, r_{|R_t|}\}$ . It should be stressed that every  $r$  will request for a SFC composed of all the VNF types in  $K$ . We will denote the  $k$ -type VNF requested by  $r$  as  $f_r^k, \forall k \in K, r \in R_t$ . Key notations for our vCDN SFC Deployment Problem are listed in Table 1.

We now enlist all the network elements and parameters that are part of the proposed optimization problem:

- $N = N_{CP} \cup N_H \cup N_{UC}$  as the set of all **nodes** of the CDN network,
- $K = \{streamer, transcoder, compressor, cache\}$  is the set of **VNF types** considered in our model,
- $L(N \times N)$ , is the set of **links** between nodes in  $N$ , so that  $(i, j) \in L, \forall i, j \in N$ ,
- $\mathbf{R} = \{CPU, Bandwidth, and Memory\}$ , is the set of **resource types** for every VNF container,
- The **resource cost** matrix  $\Gamma \in \mathbb{M}(|\mathbf{R}|, |N_H|)$ , where  $\gamma_{res,i}$  is the per-unit resource cost of resource  $res$  at node  $i$ ,
- $D \in \mathbb{M}(|N|, |N|)$ , is the **link delay** matrix so that variable  $d_{i,j} \in \mathbb{R}^+$  represents the data propagation delay between the nodes  $i$  and  $j$ . We assume  $d_{i,j} = 0$  for  $i = j$ . Notice that  $D$  is symmetric,
- $O \in \mathbb{M}(|N|, |N|)$ , is the **data-transportation costs** matrix, where  $o_{i,j} \in \mathbb{R}^+$  is the unitary data-transportation cost for the link that connects  $i$  and  $j$ . We assume  $o_{i,j} = 0$  for  $i = j$ . Notice that  $O$  is symmetric,
- $I_k, \forall k \in K$  are the parameters indicating the **instantiation times** of each  $k$ -type VNF.
- $R_t = \{r_1, r_2, \dots, r_{|R_t|}\}$  is the set of incoming **session requests** set during time-step  $t$ . Every request  $r$  is characterized by  $b_r \in [0, 1]$ , the scaled **maximum bitrate**,  $p_r \in [0, 1]$ , the mean scaled **payload**,  $l_r$ , the **content provider** requested, and  $u_r$ , the **user cluster** from which  $r$  was originated,
- $T$  is a fixed parameter indicating the maximum RTT threshold value for the incoming Live-Streaming requests.
- $P^* \in \mathbb{M}(|\mathbf{R}|, |N_H|)$  is the **optimal processing times** matrix, where  $\rho_{res,k}^*$  is the processing-time contribution of the  $res$  resource in any  $k$ -type VNF assuming optimal utilization conditions,

- $\Psi \in \mathbb{M}(|\mathbf{R}|, |N_H|)$  is the **time degradation** matrix, where  $\psi_{res,k}$  is the parameter representing the *degradation base* for the *res* resource in any *k*-type VNF,
- $c_{res,k,i}^t, \forall res \in \mathbf{R}, \forall k \in K, \forall i \in N_H$  are a set of variables indicating the *res resource capacity* of  $f_i^k$  VNF during *t*.

**Table 1.** Key notations for our vCDN SFC Deployment Problem.

Notation	Meaning
$N_{CP}$	The set of content-provider nodes
$N_H$	The set of VNF hosting nodes
$N_{UC}$	The set of user clusters
$K$	The set of VNF types (cache, compressor, transcoder, and streamer)
$t$	A fixed time-step in our simulation
$R_t$	The set of incoming SFC requests during time-step <i>t</i>
$T$	The max tolerable RTT for SFC any Live-Streaming request
$b_r$	The maximum scaled bitrate of <i>r</i>
$p_r$	The mean scaled session payload of <i>r</i>
$\hat{f}_r^k$	The <i>k</i> -type VNF requested by <i>r</i>
$l_r$	The <i>channel</i> or content provider requested by <i>r</i>
$s_r$	The <i>session workload</i> of <i>r</i>
$\gamma_{res,i}$	The unit cost of <i>res</i> resource in hosting node <i>i</i>
$d_{i,j}$	The data propagation delay between nodes <i>i</i> and <i>j</i>
$f_i^k$	The <i>k</i> -type VNF container instance in node <i>i</i>
$a_i^k$	1 if $f_i^k$ is instantiated at the beginning of <i>t</i>
$x_{r,i}^k$	1 if $\hat{f}_r^k$ is assigned to node <i>i</i> , 0 otherwise
$z_{i,j,k}^r$	1 if the link between <i>i</i> and <i>j</i> is used to reach $\hat{f}_r^k$ , 0 otherwise
$y_{l,i,k}$	1 if $f_i^k$ is currently ingesting channel <i>l</i> , 0 otherwise
$c_{res,k,i}^t$	The <i>res</i> resource provision in $f_i^k$ during <i>t</i>
$\sigma_{k,r,res}$	The client <i>res</i> resource demand of <i>r</i> in any <i>k</i> -type VNF instance
$v_{k,res}$	The <i>res</i> resource demand for content ingestion in any <i>k</i> -type VNF instance
$u_{res,k,i}$	The current <i>res</i> resource usage of $f_i^k$
$\mu_{res,k,i}$	The current <i>res</i> resource utilization of $f_i^k$
$\rho_{res,i,k}^r$	The processing time contribution of <i>res</i> resource in $f_i^k$ for $\hat{f}_r^k$
$\phi_p$	The normalization exponent for mean payload in the <i>session workload</i> formula
$\phi_b$	The normalization exponent for maximum bitrate in the <i>session workload</i> formula
$o_{i,j}$	The unitary data-transportation cost for the link between nodes <i>i</i> and <i>j</i>
$v_r$	1 if the SFC assigned to <i>r</i> respects the maximum tolerable RTT, 0 otherwise
$\rho_{res,k}^*$	The processing-time contribution of <i>res</i> in any <i>k</i> -type VNF assuming optimal utilization conditions

### 2.1.2. Optimization Statement

Given a Live-Streaming vCDN constituted by the parameters enlisted in Section 2.1.1, we must decide the SFC deployment scheme for each incoming session request *r*, considering the penalties in the resulting RTT caused by the eventual instantiation of VNF containers, cache MISS events, and over-utilization of network resources. We must also consider that the entity of the vCDN operational costs is derived from our SFC deployments. We must deploy SFCs for every request to maximize the resulting QoS and minimizing Operational Costs.

### 2.1.3. Decision Variables

We propose a discrete optimization problem: For every incoming request  $r \in R_t$ , the decision variables in our optimization problem are the binary variables  $x_{r,i}^k, \forall i \in N_H, \forall k \in K$  that equal 1 if  $\hat{f}_r^k$  is assigned to  $f_i^k$ , and 0 otherwise.

### 2.1.4. Penalty Terms and Feasibility Constraints

We model two penalty terms and two feasibility constraints for our optimization problem. The first penalty term is the **Quality of Service penalty term** and is modeled as follows. The acceptance ratio during time-step  $t$  is computed as:

$$\chi_Q^t = \frac{\sum_{r \in R_t} v_r}{|R_t|} \tag{1}$$

where the binary variable  $v_r$  indicates if the SFC assigned to  $r$  respects or not its maximum tolerable RTT, denoted by  $T_r$ :

$$v_r = \begin{cases} 1, & \text{if } RTT_r \leq T_r \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Notice that  $RTT_r$  is the round-trip-time of  $r$  and is computed as:

$$RTT_r = \sum_{i,j \in N} \sum_{k \in K} z_{i,j,k}^r \cdot d_{i,j} + \sum_{i \in N_H} \sum_{k \in K} x_{r,i}^k \cdot \{(1 - a_i^{t,k}) \cdot I_k + \rho_{i,k}^t\} \tag{3}$$

where:

- The binary variable  $z_{i,j,k}^r$  equals 1 if the link between nodes  $i$  and  $j$  is used to reach  $\hat{f}_r^k$ , and 0 otherwise,
- The parameter  $d_{i,j} \in \mathbb{R}^+$  represents the data-propagation delay between the nodes  $i$  and  $j$ . (We assume a unique path between every node  $i$  and  $j$ ),
- The binary variable  $x_{r,i}^k$  equals 1 if  $\hat{f}_r^k$  is assigned to  $f_i^k$ , and 0 otherwise,
- The binary variable  $a_i^{t,k}$  equals 1 if  $f_i^k$  is instantiated at the beginning of  $t$ , and 0 otherwise,
- $I_k$  is a parameter indicating the instantiation time of any  $k$ -type VNF.
- $\rho_{i,k}^r$  is the processing time of  $r$  in  $f_i^k$ .

Notice that, by modeling  $RTT_r$  with (3), we include data-propagation delays, processing time delays, and VNF instantiation times when needed: We will include the delay to the content provider in such RTT only in the case of a cache MISS. In other words, if  $i$  is a CP node, then  $z_{i,j,cache}^r$  will be 1 only if  $f_j^{cache}$  was not ingesting content from  $i$  at the time of receiving the assignation of  $\hat{f}_r^{cache}$ . On the other hand, whenever  $\hat{f}_r^k$  is assigned to  $f_i^k$ , but  $f_i^k$  is not instantiated at the beginning of  $t$ , then the VNO will instantiate  $f_i^k$ , but adequate delay penalties are added to  $RTT_r$ , as shown in (3). Notice that we approximate the VNF instantiation states in the following manner: Any VNF that is not instantiated during  $t$  and receives a VNF request to manage starts its own instantiation and finishes such instantiation process at the beginning of the  $t + 1$ . From that moment on, unless the VNF has been turned off in the meantime because all its managed sessions are timed out, the VNF is considered ready to manage new incoming requests without any instantiation time penalty.

Recall that we model three resource types for each VNF: CPU, Bandwidth, and Memory. We model the processing time of any  $r$  in  $f_i^k$  as the sum of the processing times related to each of these resources:

$$\rho_{i,k}^r = \rho_{cpu,i,k}^r + \rho_{mem,i,k}^r + \rho_{bw,i,k}^r \tag{4}$$

where  $\rho_{res,i,k}^r, \forall res \in \{cpu, mem, bw\}$  are each of the resource processing time contributions for  $r$  in  $f_i^k$ , and each of such contributions is computed as:

$$\rho_{res,i,k} = \begin{cases} \rho_{res,k}^* \cdot \psi_{res,k}^{\mu_{res,k,i}-1}, & \text{if } \frac{\mu_{res,k,i}}{\alpha_{res,k}} > 1 \\ \rho_{res,k}^* & \text{otherwise} \end{cases} \quad (5)$$

where:

- the parameter  $\rho_{res,k}^*$  is the processing-time contribution of  $res$  in any  $k$ -type VNF assuming optimal utilization conditions,
- $\psi_{res,k}$  is a parameter representing the *degradation base* for  $res$  in any  $k$ -type VNF,
- the variable  $\mu_{res,k,i}$  is the  $res$  utilization in  $f_i^k$  at the moment when  $f_i^k$  is assigned.

Note that (5) models utilization-dependent processing times. The resource utilization in any  $f_i^k$ , denoted as  $\mu_{res,k,i}, \forall res \in \{cpu, mem, bw\}$  is computed as:

$$\mu_{res,k,i} = \begin{cases} \frac{u_{res,k,i}}{c_{res,k,i}^t}, & \text{if } c_{res,k,i} > 0 \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

where  $u_{res,k,i}$  is the instantaneous  $res$  resource usage in  $f_i^k$ , and  $c_{res,k,i}^t$  is the  $res$  resource capacity of  $f_i^k$  during  $t$ . The value of  $c_{res,k,i}^t$  is fixed during an entire time-step  $t$  and depends on any dynamic resource provisioning algorithm acted by the VNO. In this work we assume a bounded *greedy* resource provisioning policy as specified in Appendix A.1. On the other hand, if we denote with  $\bar{R}_t$  the a subset of  $R_t$  that contains the requests that have already been accepted at the current moment, we can compute  $u_{res,k,i}$  as:

$$u_{res,k,i} = \hat{u}_{res,k,i}^t + \sum_{r \in \bar{R}_t} x_{k,r,i} \cdot \sigma_{k,r,res} + \sum_{l \in N_{CP}} y_{l,i}^k \cdot v_{k,res} \quad (7)$$

where:

- The variable  $\hat{u}_{res,k,i}^t$  indicates the  $res$  resource demand in  $f_i^k$  at the beginning of time-step  $t$ ,
- The binary variable  $x_{k,r,i}$  was already defined and it indicates if  $f_r^k$  is assigned to  $f_i^k$ ,
- $\sigma_{k,r,res}$  is the  $res$  resource demand faced by any  $k$ -type VNF when serving  $r$ , and we call it the *client resource-demand*,
- The binary variable  $y_{l,i}^k$  is 1 if  $f_i^k$  is currently ingesting content from content provider  $l$ , and 0 otherwise,
- The parameter  $v_{k,res}$  models the  $res$  resource demand faced by any  $k$ -type VNF when ingesting content from any content provider.

Notice that, modeling resource usage with (7), we take into account not only the resource demand associated with the content transmission, but we also model the resource usage related to each content ingestion task the VNF is currently executing.

The  $res$  resource demand that any  $k$ -type VNF faces when serving a session request  $r$  is computed as:

$$\sigma_{k,r,res} = \sigma_{max,k,res} \cdot s_r \quad (8)$$

where  $\sigma_{max,k,res}$  is a fixed parameter that indicates the maximum possible  $res$  resource consumption implied while serving any session request incoming to any  $k$ -type VNF. The variable  $s_r \in [0, 1]$  instead, is indicating the *session workload* of  $r$ , which depends on the specific characteristics of  $r$ . In particular, the *session workload* will depend on the *normalized maximum bitrate* and the *mean payload per time-step* of  $r$ , denoted as  $b_r$  and  $p_r$ , respectively:

$$s_r = (p_r)^{\phi_p} \cdot (b_r)^{\phi_b} \quad (9)$$

In (8), the parameters  $\phi_p, \phi_b \in [0, 1]$  do not depend on  $r$  and are fixed normalization exponents that balance the contribution of  $b_r$  and  $p_r$  in  $s_r$ .

Recall that the binary variable  $v_r$  indicates if the SFC assigned to  $r$  respects or not its maximum tolerable RTT. Notice that we can assess the total throughput served by the vCDN during  $t$  as:

$$\chi_T^t = \chi_Q^t \cdot \sum_{r \in R_t} s_r \tag{10}$$

The second penalty term is related to the **Operational Costs**, which is constituted by both the hosting costs and the Data-transportation costs. We can compute the *Hosting Costs* for our vCDN during  $t$  as:

$$\chi_H^t = \chi_H^{t-1} - \tilde{\chi}_H^t + \sum_{i \in N_H} \sum_{k \in K} \sum_{res \in \mathbf{R}} \gamma_{res,i} \cdot c_{res,k,i}^t \tag{11}$$

where

- $\chi_H^{t-1}$  are the total Hosting Costs at the end of time-step  $t - 1$ ,
- $\tilde{\chi}_H^t$  are the hosting costs related to the timed-out sessions at the beginning of time-step  $t$ ,
- $\mathbf{R}$  is the set of resources we model, i.e., Bandwidth, Memory, and CPU,
- $\gamma_{res,i}$  is the per-unit resource cost of resource  $res$  at node  $i$ .

Recall that  $c_{res,k,i}^t$  is the  $res$  resource capacity at  $f_i^k$  during  $t$ . Notice that different nodes may have different per-unit resource costs as they may be instantiated in different cloud providers. Thus, modeling the hosting costs using (11), we have considered a possible multi-cloud vCDN deployment. Notice also that, using (11), we keep track of the current total hosting costs for our vCDN assuming that timed-out session resources are released at the end of each time-step.

We now model the Data-Transportation Costs. In our vCDN model, each hosting node instantiates a maximum of one VNF of each type. Consequently, all the SFCs that exploit the same link for transferring the same content between the same pair of VNFs will exploit a unique connection. Therefore, to realistically assess DT costs, we create the notion of session DT *amortized-cost*:

$$d_{cost}^r = \sum_{i,j \in N_H} \sum_{k \in K} \frac{p_r \cdot z_{i,j,k}^r \cdot o_{i,j}}{|R_r^{(i,j,k)}|} \tag{12}$$

where  $o_{i,j}$  is a parameter indicating the unitary DT cost for the link between  $i$  and  $j$ , and  $R_r^{(i,j,k)}$  is the set of SFCs that are using the link between  $i$  and  $j$  to transmit to  $f_j^k$  the content related to the same CP requested by  $r$ . Notice that DT costs for  $r$  are proportional to the mean payload  $p_r$ . Recall that  $z_{i,j,k}^r$  indicates if the link between  $i$  and  $j$  is used to reach  $f_j^k$ . According to (12), we compute the session DT cost for any session request  $r$  in the following manner: For each link on our vCDN, we first compute the whole DT cost among such a link. We then compute the number of concurrent sessions that are using such a link for transferring the same content requested by  $r$ . Lastly, we compute the ratio between these quantities and sum such ratios for every hop in the SFC of  $r$  to obtain the whole session amortized DT cost. The total amortized DT costs during  $t$  are then computed as:

$$\chi_D^t = \chi_D^{t-1} - \tilde{\chi}_D^t + \sum_{r \in R_t} v_r \cdot d_{cost}^r \tag{13}$$

where

- $\chi_D^{t-1}$  are the total DT costs at the end of the  $t - 1$  time-step,
- $\tilde{\chi}_D^t$  are the total DT costs regarding the timed-out sessions at the beginning of time-step  $t$ ,
- $d_{cost}^r$  is the session DT cost for  $r$  computed with (12). Recall that  $v_r$  indicates if  $r$  was accepted or not based on its resultant RTT.

On the other hand, the first constraint is the **VNF assignment constraint**: For any live-streaming request  $r$ , every  $k$ -type VNF request  $\hat{f}_r^k$  must be assigned to one and only one node in  $N_H$ . We express such a constraint follows:

$$\sum_{i \in N_H} x_{r,i}^k = 1, \forall r \in R_t, \forall k \in K, \tag{14}$$

Finally, the second constraint is the **minimum service constraint**. For any time-step  $t$ , the acceptance ratio must be greater or equals than 0.5. We express such a constraint as:

$$\chi_Q^t \geq 0.5, \forall t \in \mathbb{N} \tag{15}$$

One could optimize operational costs by discarding a significant percentage of the incoming requests instead of serving them. The fewer requests are served, the less the resource consumption entity and the hosting costs will be. Also, data transfer costs are reduced when less traffic is generated due to the rejection of live-streaming requests. However, the constraint in (15) is created to avoid such naive solutions to our optimization problem.

### 2.1.5. Optimization Objective

We model a multi-objective SFC deployment optimization: At each simulation time-step  $t$ , we measure the accomplishment of three objectives:

- Our first goal is to maximize the network throughput as defined in (10), and we express such objective as  $\max(\chi_T^t)$ ,
- Our second goal is to minimize the hosting costs as defined in (11), and we express such objective as  $\min(\chi_H^t)$ ,
- Our third goal is to minimize the DT cost as defined in (13), and such objective can be expressed as  $\min(\chi_D^t)$ .

We tackle such a multi-objective optimization goal with a weighted-sum method that leads to a single objective function:

$$\max(w_T \cdot \chi_T^t - w_D \cdot \chi_D^t - w_H \cdot \chi_H^t) \tag{16}$$

where  $w_T$ ,  $w_H$ , and  $w_D$  are parametric weights for the network throughput, hosting costs, and data transfer costs, respectively.

### 2.2. Proposed Solution: Deep Reinforcement Learning

Any RL framework is composed of an optimization objective, a reward policy, an environment, and an agent. In RL scenarios, a Markov Decision Process (MDP) is modeled, where the Environment conditions are the nodes of a Markov Chain (MC) and are generally referred to as *state-space* samples. The agent iteratively observes the state-space and chooses actions from the *action-space* to interact with the Environment. Each action is corresponded by a *reward* signal and leads to the *transition* to another node in the Markov Chain, i.e., to another point in the *state-space*. Reward signals are generated by a *action-reward* that drives learning towards optimal action *policies* under dynamic environment conditions.

In this work, we propose a DRL-based framework to solve our Online SFC Deployment problem. To do that, we first need to embed our optimization problem in a Markov Decision Process (MDP). We then need to create an *action reward mechanism* that drives the agent to learn optimal SFC Deployment policies, and finally, we need to specify the DRL algorithm we will use for solving the problem. The transition between states of the MDP will be indexed by  $\tau, \forall \tau \in \mathbb{N}$  in the rest of this paper.

#### 2.2.1. Embedding SFC Deployment Problem into a Markov Decision Process

Following [14,32,48,51,52], we propose to serialize the SFC Deployment process into single-VNF assignment actions. In other words, our agent interacts with the Environment

each time a particular VNF request,  $f_r^k$ , has to be assigned to a particular VNF instance,  $f_i^k$  of some hosting node  $i$  in the vCDN. Consequently, the actions of our agent, denoted by  $a_\tau$  are the single-VNF assignment decisions for each VNF request of a SFC.

Before taking any action, the agent observes the environment's conditions. We propose to embed such conditions onto a vector  $s_\tau$  that contains a snapshot of the current incoming request and hosting nodes' conditions. In particular,  $s_\tau$  will be formed by the concatenation of three vectors:

$$s_\tau = (\hat{R}, \hat{I}, \hat{U}) \tag{17}$$

where:

- $\hat{R} = (l_r, u_r, s_r, f_r^-)$  is called the request vector, and contains information about the VNF request under assignment. In particular,  $\hat{R}$  codifies the requested CP,  $l_r$ , the client cluster from which the request was originated,  $u_r$ , the request's session workload,  $s_r$ , and the number of pending VNFs to complete the deployment of the current SFC,  $f_r^-$ . Notice that  $f_r^- \in \{0, 1, 2, 3, 4\}$  and that  $f_r^-$  goes each time from 4 to 0 as our agent performs the assignments actions regarding  $r$ . Note that the first three components of  $\hat{R}^\tau$  are invariable for the whole set of VNF requests regarding the SFC of  $r$ . Notice also that, using a *one-hot* encoding for  $l_r$ ,  $u_r$ , and  $f_r^-$ , the dimension of  $\hat{R}$  is  $|\hat{R}| = |N_{UC}| + |N_{CP}| + |K| + 1$ ,
- $\hat{I} = (\hat{i}_1, \hat{i}_2, \dots, \hat{i}_{|N_H|})$  is a binary vector where  $\hat{i}_j$  equals 1 if  $f_j^k$  is ingesting the CP requested by  $r$  and 0 otherwise,
- $\hat{U} = (\hat{u}_1, \hat{u}_2, \dots, \hat{u}_{|N_H|})$ , where  $\hat{u}_i$  is the utilization value of the resource that is currently the most utilized resource in  $f_k^i$ .

### 2.2.2. Action-Reward Schema

When designing the action-reward schema, we take extreme care in giving the right entity to the penalty of resource over-utilization, as it seriously affects QoS. We also include a cost-related penalty to our reward function to jointly optimize QoS and Operational Costs. Recall that the actions taken by our agent are the single-VNF assignment decisions for each VNF request of a SFC. At each iteration  $\tau$ , our agent observes the state information  $s_\tau$ , takes an action  $a_\tau$ , and receives a correspondent reward  $r(s_\tau, a_\tau)$  computed as:

$$r(s_\tau, a_\tau) = w_Q \cdot r_{QoS}(s_\tau, a_\tau) - v \cdot (w_D \cdot \chi_D^t + w_H \cdot \chi_H^t) \tag{18}$$

where:

- The parameters  $w_Q$ ,  $w_D$ , and  $w_H$  are weights for the reward contributions relating to the QoS, the DT costs, and the Hosting Costs, respectively,
- $r_{QoS}(s_\tau, a_\tau)$  is the reward contribution that is related to the QoS optimization objective,
- $v$  is a binary variable that equals 1 if action  $a$  corresponds to the last assignment step of the last session request arrived in the current simulation time-step  $t$ , and 0 otherwise.

Recall that  $\chi_D^t$  and  $\chi_H^t$  are the total DT costs and hosting costs of our vCDN at the end of the simulation time-step  $t$ . Using (18), we subtract a penalty proportional to the current whole hosting and DT costs in the vCDN only at the last transition of each simulator time-step, i.e., when we assign the last VNF of the last SFC in  $R_t$ . Such a sparse cost penalty was also proposed in [14].

When modeling the QoS-related contribution of the reward instead, we propose the usage an *inner delay-penalty function*, denoted as  $d(t)$ . In practice,  $d(t)$  will be continuous and non-increasing. We design  $d(t)$  in such a way that  $d(t) < 0, \forall t > T$ . Recall that  $T$  is a fixed parameter indicating the maximum RTT threshold value for the incoming Live-Streaming requests. We specify the inner delay-penalty function used in our simulations in Appendix A.2.

Whenever our agent performs an assignment action  $a$ , for a VNF request  $f_r^k$  in  $r$ , we compute the generated contribution to the RTT of  $r$ . In particular, we compute the processing time of  $r$  in the assigned VNF, eventual instantiation times, and the transmission

delay to the chosen node. We sum such RTT contribution at each assignation step to form the current partial RTT, which we denote as  $t_r^a$ . The QoS-related part of the reward assigned to  $a$  is then computed as:

$$r_{QoS}(a) = \begin{cases} d(t_r^a) \cdot 2^{-\hat{f}_r^-}, & \text{if } \hat{f}_r^- > 0 \text{ and } d(t_r^a) > 0 \\ d(t_r^a) \cdot 2^{\hat{f}_r^-}, & \text{if } \hat{f}_r^- > 0 \text{ and } d(t_r^a) < 0 \\ 1, & \text{if } \hat{f}_r^- = 0 \text{ and } d(t_r^a) > 0 \\ 0, & \text{if } \hat{f}_r^- = 0 \text{ and } d(t_r^a) < 0 \end{cases} \quad (19)$$

If we look at the first line of (19), we realize that a positive reward is given for every assignment that results in a non-prohibitive partial RTT. Moreover, such a positive reward is inversely proportional to  $\hat{f}_r^-$  (the number of pending assignations for the complete deployment of the SFC of  $r$ ). Notice that, since  $t_r^a$  is cumulative, we give larger rewards to the latter assignation actions of an SFC, as it is more difficult to avoid surpassing the RTT limit at the end of the SFC deployment with respect to the beginning.

The second line in (19) shows instead that a negative reward is given to the agent whenever  $t_r^a$  exceeds  $T$ . Further, such a negative reward worsens proportionally to the prematurity of the assignation action that caused  $t_r^a$  to surpass  $T$ . Such a worsening makes sense because bad assignation actions are easier to occur at the end of the SFC assignation process with respect to the beginning.

Finally, the third and fourth lines in (19) correspond to the case when we the agent performs the last assignation action of an SFC. The third line indicates that the QoS related reward is equal to 1 whenever a complete SFC request  $r$  is deployed, i.e., when every  $\hat{f}_r^k$  in the SFC of  $r$  has been assigned without exceeding the RTT limit  $T$ , and the last line tells us that the reward will be 0 whenever the last assignation action incurs in a non-acceptable RTT for  $r$ .

This reward schema is the main contribution of our work. According to the MDP embedding proposed Section 2.2.1, the majority of actions taken by our agent are given a non-zero reward. Such a reward mechanism is called *dense*, and it improves the training convergence to optimal policies. Notice also that, in contrast to [14], our reward mechanism penalizes bad assignments in contrast to ignoring them. Such an inclusion enhances the agent exploration in the action space and reduces the possibility of converging to local optima.

### 2.2.3. DRL Algorithm

Any RL agent receives a reward  $R_\tau$  for each action taken,  $a_\tau$ . The function that RL algorithms seek to maximize is called the *discounted future reward* and is defined as:

$$G_\tau = R_{\tau+1} + \gamma R_{\tau+2} + \dots = \sum_{k=0}^{\infty} \gamma^k R_{\tau+k+1} \quad (20)$$

where  $\gamma$  is a fixed parameter known as the *discount factor*. The objective is then to learn an action *policy* that maximizes such the discounted future reward. Given a specific policy  $\pi$ , the *action value function*, also called *Q-value function* indicates how much valuable it is to take a specific action  $a_\tau$  being at state  $s_\tau$ :

$$Q_\pi(s, a) = E_\pi[G_\tau | s_\tau = s, a_\tau = a] \quad (21)$$

from (21) we can derive the *recursive Bellman equation*:

$$Q(s_\tau, a_\tau) = R_{\tau+1} + \gamma Q(s_{\tau+1}, a_{\tau+1}) \quad (22)$$

Notice that if we denote the final state with  $s_{final}$ , then  $Q(s_{final}, a) = R_a$ . The Temporal Difference learning mechanism uses (22) to approximate the Q-values for state-action pairs

in the traditional *Q-learning algorithm*. However, in large state or action spaces, it is not always feasible to use tabular methods to approximate the Q-values.

To overcome the traditional Q-learning limitations, Mnih et al. [53] proposed the usage of a Deep Artificial Neural Network (ANN) approximator of the Q-value function. To evict convergence to local-optima, they proposed to use an  $\epsilon$ -greedy policy where actions are sampled from the ANN with probability  $1 - \epsilon$  and from a random distribution with probability  $\epsilon$ , where  $\epsilon$  decays slowly at each MDP transition during training. They also used the *Experience Replay* (ER) mechanism: a data structure  $\mathcal{D}$  keeps  $(s_\tau, a_\tau, r_\tau, s_{\tau+1})$  transitions for sampling uncorrelated training data and improve learning stability. ER mitigates the high correlation presented in sequences of observations during online learning. Moreover, authors in [54] implemented two neural network approximators for (21), the Q-network and the Target Q-network, indicated by  $Q(s, a, \theta)$  and  $Q(s, a, \theta^-)$ , respectively. In [54], the target network is updated only periodically to reduce the variance of the target values and further stabilize learning with respect to [53]. Authors in [54] use stochastic gradient descent to minimize the following loss function:

$$\mathcal{L}(\theta) = E_{(s_\tau, a_\tau, r_\tau, s_{\tau+1}) \sim U(\mathcal{D})} \{ [r + \gamma \max_a Q(s_\tau, a, \theta^-) - Q(s_\tau, a_\tau; \theta)]^2 \} \tag{23}$$

where minimization of (23) is done with respect to the parameters of  $Q(s, a, \theta)$ . Van Hasselt et al. [55] applied the concepts of Double Q-Learning [56] on large-scale function approximators. They replaced the target value in (23) with a more sophisticated target value:

$$\mathcal{L}(\theta) = E_{(s_\tau, a_\tau, r_\tau, s_{\tau+1}) \sim U(\mathcal{D})} \{ [r_\tau + \gamma Q(s_{\tau+1}, \operatorname{argmax}_a Q(s_{\tau+1}, a; \theta), \theta^-) - Q(s_\tau, a_\tau; \theta)]^2 \} \tag{24}$$

Doing such a replacement, authors in [55] avoided over-estimations of the Q-values which characterized (23). This technique is called Double Deep Q-Learning (DDQN), and it also helps to decorrelate the noise introduced by  $\theta$ , from the noise of  $\theta^-$ . Notice that  $\theta$  are the parameters that approximate the function used to choose the best actions, while  $\theta^-$  are the parameters of the approximator used to evaluate the choices. Such a differentiation in the *learning* and *acting* policies is also called *off-policy* learning.

Instead, Wang et al. [41] proposed a change in the architecture of the ANN approximator of the Q-function: they used a decomposition of the action value function in the sum of two other functions: the *action-advantage function* and the *state-value function*:

$$Q_\pi(s, a) = V_\pi(s) + A_\pi(a) \tag{25}$$

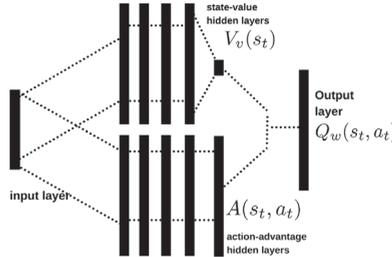
Authors in [41] proposed a two-stream architecture for an ANN approximator, where one stream approximated  $A_\pi$  and the other approximated  $V_\pi$ . They integrate such contributions at the final layer of the ANN  $Q_\pi$  using:

$$Q(s, a; \theta_1, \theta_2, \theta_3) = V(s; \theta_1, \theta_3) + (A(s, a; \theta_1, \theta_2) - \frac{1}{|\mathcal{A}|} \sum_{a'} A(s, a'; \theta_1, \theta_2)) \tag{26}$$

where  $\theta_1$  are the parameters of the first layers of the ANN approximator, while  $\theta_2$  and  $\theta_3$  are the parameters encoding the action-advantage and the state-value heads, respectively. This architectural innovation works as an attention mechanism for states where actions have more relevance with respect to other states and is known as *Dueling DQN*. Dueling architectures have the ability to generalize learning in the presence of many similar-valued actions.

For our SFC Deployment problem, we propose the usage of the DDQN algorithm [55] where the ANN approximator of the Q-value function uses the dueling mechanism as in [41]. Each layer of our Q-value function approximator is a fully connected layer. Consequently, it can be classified as a multilayer Perceptron (MLP) even if it has a two-stream architecture. Even if we approximate  $A_\pi(a)$  and  $V_\pi(s)$  with two streams, the final output layer of our ANN approximates the Q-value for each action using (26). The input neu-

rons receive the state-space vectors  $s_\tau$  specified in Section 2.2.1. Figure 2 schematizes the proposed topology for our ANN. The parameters of our model are detailed instead in Table 2.



**Figure 2.** Dueling-architected DDQN topology for our SFC Deployment agent: A two-stream deep neural network. One stream approximates the state-value function, and the other approximates the action advantage function. These values are combined to get the state-action value estimation in the output layer. The inputs are instead the action is taken and the current state.

**Table 2.** Deep ANN Assigner topology Parameters.

Parameter	Value
Action-advantage hidden layers	2
State-value hidden layers	2
hidden layers dimension	128
Input layer dimension	$2 \cdot  N_H  + ( N_{UC}  +  N_{CP}  +  K  + 1)$
Output layer dimension	$ N_H $
Activation function between hidden layers	ReLU

We index the training episodes with  $e \in [0, 1, \dots, M]$ , where  $M$  is a fixed training hyper-parameter. We assume that an episode ends when all the requests of a fixed number of simulation time-steps  $N_{ep}$  have been processed. Notice that each simulation time-step  $t$  may have a different number of incoming requests,  $|R_t|$ , and that every incoming request  $r$  will be mapped to an SFC of length  $|K|$ , which coincides with the number of MDP transitions on each SFC deployment process. Consequently, the number of transitions in an episode  $e$  will be then given by

$$N^e = \sum_{t \in [t_0^e, t_f^e]} |K| \cdot |R_t| \tag{27}$$

where  $t_0^e = t \cdot N_{ep}$  and  $t_f^e = t \cdot (N_{ep} + 1)$  are the initial and final simulation timesteps of episode  $e$ , respectively (Recall that  $t \in \mathbb{N}$ ).

To improve training performance and avoid convergence to local optima, we use the  $\epsilon$ -greedy mechanism. We introduce a high number of randomly chosen actions at the beginning of our training phase and progressively diminish the probability of taking such random actions. Such randomness should help to reduce the bias in the initialization of the ANN approximator parameters. In order to gradually lower the number of random moves as our agent learns the optimal policy, our  $\epsilon$ -greedy policy is characterized by an exponentially decaying  $\epsilon$  as:

$$\epsilon(\tau) = \epsilon_{final} + (\epsilon_0 - \epsilon_{final}) \cdot e^{\frac{-\tau}{\epsilon_{decay}}}, \forall \tau \in \mathbb{N}^+ \tag{28}$$

where we define  $\epsilon_0$ ,  $\epsilon_{final}$ , and  $\epsilon_{decay}$  as fixed hyper-parameters such that

$$\epsilon_{decay} \gg 1 \geq \epsilon_0 \gg \epsilon_{final}$$

Notice that  $\epsilon(0) = \epsilon_0$  and

$$\lim_{\tau \rightarrow +\infty} \epsilon(\tau) = \epsilon_{final}$$

We call our algorithm Enhanced-Exploration Dense-Reward Duelling DDQN (E2-D4QN) SFC Deployment. Algorithm 1 describes the training procedure of our E2-D4QN DRL agent. We call *learning network* the ANN approximator used to choose actions. In lines 1 to 3, we initialize the replay memory, the parameters of the first layers ( $\theta_1$ ), the action-advantage head ( $\theta_2$ ), and the state-value head ( $\theta_3$ ) of the ANN approximator. We then initialize the target network with the same parameter values of the learning network. We train our agent for  $M$  epochs, each of which will contain  $N_e$  MDP transitions. In lines 6–10 we set an *ending episode signal*  $\tau_{end}$ . We need such a signal because, when the final state of an episode has been reached, the loss should be computed with respect to the pure reward of the last action taken, by definition of  $Q(s, a)$ . At each training iteration, our agent observes the environment conditions, takes an action using the  $\epsilon$ -greedy mechanism, obtains a correspondent reward, and transits to another state (lines 11–14). Our agent stores the transition in the replay buffer and then randomly samples a batch of stored transitions to run the stochastic gradient descent on the loss function in (24) (lines 14–25). Notice that the target network will only be updated with the parameter values of the learning value each  $U$  iterations to increase training stability, where  $U$  is a fixed hyper-parameter. The complete list of the training hyper-parameters used for training is enlisted in Appendix A.4.

---

**Algorithm 1** E2-D4QN.

---

- 1: Initialize  $\mathcal{D}$
  - 2: Initialize  $\theta_1, \theta_2$ , and  $\theta_3$  randomly
  - 3: Initialize  $\theta_1^-, \theta_2^-$ , and  $\theta_3^-$  with the values of  $\theta_1, \theta_2$ , and  $\theta_3$ , respectively
  - 4: **for** episode  $e \in \{1, 2, \dots, M\}$  **do**
  - 5:     **while**  $\tau \leq N^e$  **do**
  - 6:         **if**  $\tau = N^e$  **then**
  - 7:              $\tau_{end} \leftarrow True$
  - 8:         **else**
  - 9:              $\tau_{end} \leftarrow False$
  - 10:         **end if**
  - 11:         Observe state  $s_\tau$  from simulator.
  - 12:         Update  $\epsilon$  using (28).
  - 13:         Sample a random assignation  $a_t$  action with probability  $\epsilon$   
or  $a_\tau \leftarrow \underset{a}{\operatorname{argmax}} Q(s_\tau, a; \Theta)$  with probability  $1 - \epsilon$ .
  - 14:         Obtain the reward  $r_\tau$  using (18),  
and the next state  $s_{\tau+1}$  from the environment.
  - 15:         Store transition tuple  $(s_\tau, a_\tau, r_\tau, s_{\tau+1}, \tau_{end})$  in  $\mathcal{D}$ .
  - 16:         Sample a batch of transition tuples  $\mathcal{T}$  from  $\mathcal{D}$ .
  - 17:         **for all**  $(s_j, a_j, r_j, s_{j+1}, \tau_{end}) \in \mathcal{T}$  **do**
  - 18:             **if**  $\tau_{end} = True$  **then**
  - 19:                  $y_j \leftarrow r_j$
  - 20:             **else**
  - 21:                  $y_j \leftarrow r + \gamma Q(s_{j+1}, \underset{a}{\operatorname{argmax}} Q(s_{j+1}, a; \theta), \theta^-)$
  - 22:             **end if**
  - 23:             Compute the temporal difference error  $\mathcal{L}(\theta)$  using (24).
  - 24:             Compute the loss gradient  $\nabla \mathcal{L}(\theta)$ .
  - 25:              $\Theta \leftarrow \Theta - l_r \cdot \nabla \mathcal{L}(\theta)$
  - 26:             Update  $\Theta^- \leftarrow \Theta$  only every  $U$  steps.
  - 27:         **end for**
  - 28:     **end while**
  - 29: **end for**
-

### 2.3. Experiment Specifications

#### 2.3.1. Network Topology

We used a real-world dataset to construct a trace-driven simulation for our experiment. We consider the topology of the proprietary CDN of an Italian Video Delivery operator in our experiments. Such an operator delivers Live video from content providers distributed around the globe to clients located in the Italian territory. This operator's network consists of 41 CP nodes, 16 hosting nodes, and 4 client cluster nodes. The hosting nodes and the client clusters are distributed in the Italian territory, while CP nodes are distributed worldwide. Each client cluster emits approximately  $1 \times 10^4$  Live-Video requests per minute. The operator gave us access to the access log files concerning service from 25–29 July 2017.

#### 2.3.2. Simulation Parameters

We took data from the first four days for training our SFC Deployment agent and used the last day's trace for evaluation purposes. Given a fixed simulation time-step interval of 15 seconds and a fixed number of  $N^e = 80$  time-steps per episode, we trained our agent for 576 episodes, which correspond to 2 runs of the 4-day training trace. At any moment, the vCDN conditions are composed by the VNF instantiation states, the caching VNF memory states, the container resource provision, utilization, etc. In the test phase of every algorithm, we should fix the initial network conditions to reduce evaluation bias. However, setting the initial network conditions like those encountered at the end of its training cycle might also bias the evaluation of any DRL agent. We want to evaluate every agent's capacity to recover the steady state from general environment conditions. Such an evaluation needs initial conditions to be different with respect to the steady-state achieved during training. In every experiment we did, we set the initial vCDN conditions as those registered at the end of the fourth day when considering a greedy SFC deployment policy. We fix the QoS, Hosting costs, and DT-cost weight parameters in (16) to 0.6, 0.3, and 0.1, respectively.

In the context of this research, we did not have access to any information related to the data-transmission delays. Thus, for our experimentation, we have randomly generated fixed data-transmission delays considering the following realistic assumptions. We assume that the delay between a content provider and a hosting node is generally bigger concerning the delay between any two hosting nodes. We also assumed that the delay between two hosting nodes is usually bigger than between hosting and client-cluster nodes. Consequently, in our experiment, delays between CP nodes and hosting nodes were generated uniformly in the interval 120–700 [ms], delays between hosting nodes, from the interval 20–250 [ms], the delays between hosting nodes and client clusters were randomly sampled from the interval 8–80 [ms]. Also, the unitary data-transportation costs were randomly generated for resembling a multi-cloud deployment scenario. For links between CP nodes and hosting nodes, we assume that unitary DT costs range between 0.088 and 0.1 USD per GB (<https://cloud.google.com/cdn/pricing> (accessed on 26 October 2021)). For links between hosting nodes, the unit DT costs were randomly generated between 0.08 and 0.004 USD per GB, while DT Cost between hosting nodes and client cluster nodes is assumed null.

The rest of the simulation parameters are given in Appendix A.3.

#### 2.3.3. Simulation Environment

The training and evaluation procedures for our experiment were made on a *Google Colab-Pro* hardware-accelerated Environment equipped with a Tesla P100-PCIE-16GB GPU, an Intel(R) Xeon(R) CPU @ 2.30GHz processor with two threads, and 13 GB of main memory. The source code for our vCDN simulator and our DRL framework's training cycles was made in python v. 3.6.9. We used torch library v. 1.7.0+cu101 (PyTorch) as a deep learning framework. The whole code is available online on our public repository ([https://github.com/QwertyJacob/e2d4qn\\_vcdn\\_sfc\\_deployment](https://github.com/QwertyJacob/e2d4qn_vcdn_sfc_deployment) (accessed on 26 October 2021)).

### 2.3.4. Compared State-of-Art Algorithms

We compare our algorithm with the NFVDeep framework presented in [14]. We have created three progressive enhancements of the NFVDeep algorithm for an exhaustive comparison with E2-D4QN. NFVDeep is a policy gradient DRL framework for maximizing network throughput and minimizing operational costs on general-case SFC deployment. Xiao et al. design a *backtracking* method: if a resource shortage or exceeded latency event occurs during SFC deployment, the controller ignores the request, and no reward is given to the agent. Consequently, sparse rewards characterize NFVDeep. The first algorithm we compare with is a reproduction of NFVDeep on our particular Live-Streaming vCDN Environment. The second algorithm introduces our dense-reward scheme on the NFVDeep framework, and we call it NFVDeep-Dense. The third method is an adaptation of NFVDeep that introduces our dueling DDQN framework but keeps the same reward policy as the original algorithm in [14], and we call it NFVDeep-D3QN. The fourth algorithm is called NFVDeep-Dense-D3QN, and it adds our dense reward policies to NFVDeep-D3QN. Notice that the difference between NFVDeep-Dense-D3QN and our E2-D4QN algorithm is that the latter does not use the backtracking mechanism: In contrast to any of the compared algorithms, we permit our agent to do wrong VNF assignments and to learn from its mistakes to escape from local optima.

Finally, we also compare our proposed algorithm with a greedy-policy lowest-latency and lowest-cost (GP-LLC) assignment agent, based on the work presented in [57]. GP-LLC is an extension of the algorithm in [57], that includes server-utilization, channel-ingestion state, and resource-costs awareness in the decisions of a greedy policy. For each incoming VNF request, GP-LLC will assign a hosting node. This greedy policy will try not to overload nodes with assignment actions and always choose the best available actions in terms of QoS. Moreover, given a set of candidate nodes respecting such a greedy QoS-preserving criterion, the LLC criterion will tend to optimize hosting costs. Appendix B describes in detail the GP-LLC SFC Deployment algorithm.

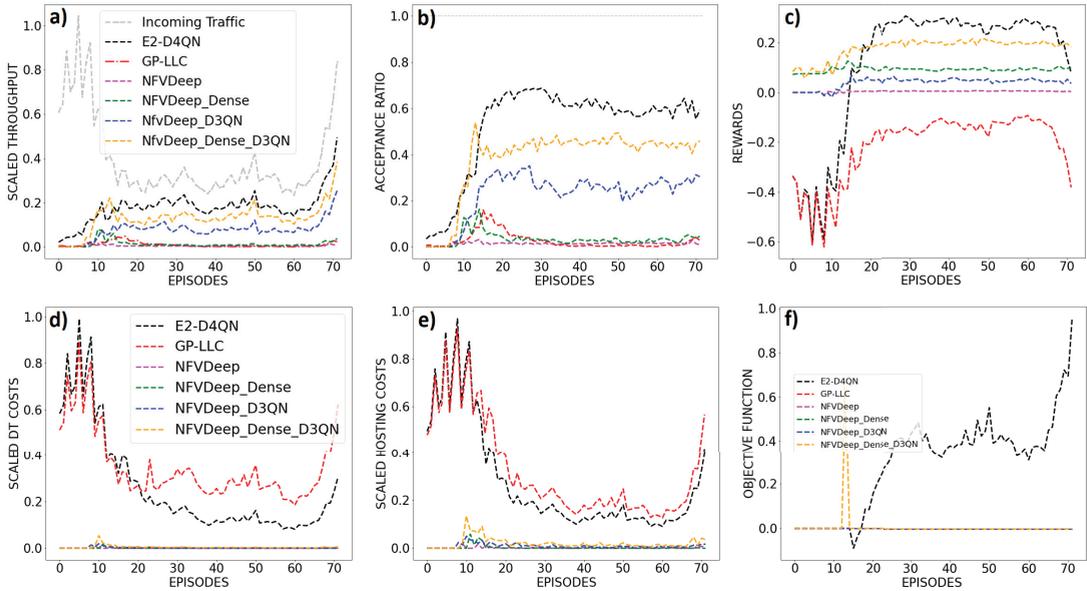
## 3. Results

Various performance metrics for all the algorithms mentioned in Section 2.3.4 are presented in Figure 3. Recall that the measurements in such a figure are taken during the 1-day evaluation trace as mentioned in Section 2.3.2. Notice that, given the time-step duration and number of time-steps per episode specified in Section 2.3.2, one-day trace consists of 72 episodes, starting at 00:00:00 h at finishing at 23:59:59 of the 29 July 2017.

### 3.1. Mean Scaled Network Throughput per Episode

The network throughput for each simulation time-step was computed using (10) and the mean values for each episode were scaled and plotted in Figure 3a. Also the scaled incoming traffic amount is plotted in such a figure. In the first twenty episodes of the trace, which correspond to the period from 0:00 to 6:00, the incoming traffic goes from intense to moderate. Incoming traffic has minor oscillations with respect to the antecedent descent from episode 20 to episode 60, and it starts to grow again from the sixtieth episode on, which corresponds to the period from 18:00 to the end of the trace.

The initial ten episodes are characterized for a comparable throughput between GP-LLC, E2-D4QN, and NFVDeep-Dense-D3QN. We can see, however, from the 20th episode on, the throughput of policy-based NFVDeep variants is lowered. From episode 15, however, which corresponds to the period from 05:00 to the end of the day, the throughput of our proposed algorithm is superior throughput with respect to every other algorithm.



**Figure 3.** Basic evaluation metrics of *E2-D4QN*, *GP-LLC*, *NFVDeep* and three variants of the latter, presented in Section 2.3.4. (a) Scaled mean network throughput per episode. (b) Mean Acceptance Ratio per episode. (c) Mean rewards per episode. (d) Mean scaled total Data-Transportation Costs per episode (e) Mean scaled total hosting costs per episode. (f) Mean scaled optimization objective per episode.

### 3.2. Mean Acceptance Ratio per Episode

The AR for each simulation time-step was computed using (1) and the mean values for each episode are plotted in Figure 3b. At the beginning of the test, corresponding to the first five episodes, E2-D4QN has a superior AR performance. From episodes 5 to 15, only E2-D4QN and NFVDeep-Dense-D3QN keep growing in the acceptance ratio. The unique algorithm that holds a satisfactory acceptance ratio during the rest of the day is E2-D4QN instead. It should be stressed that the AR cannot be one in our experiments because the parameter configuration described in Section 2.3.2 resembles overloaded network conditions on purpose.

### 3.3. Mean Rewards per Episode

Figure 3c shows the mean rewards per episode. We plot the rewards obtained at every  $|K|$  assignment steps. Notice that such a selection corresponds to the non-null rewards in the sparse-reward models.

During the first 15 episodes, -00:00 to 05:00- both GP-LLC and E2-D4QN increment their mean rewards starting from a worse performance with respect to the NFVDeep algorithms. This is explained because the *Enhanced-Exploration* mechanism of E2-D4QN and GP-LLC is the unique that includes negative rewards in the reward assignment policy. From episode 15 to 20, E2-D4QN reaches the rewards obtained by NFVDeep-Dense-D3QN, and from episode 20 on, E2-D4QN has a better performance with respect to the NFVDeep variants, most of all due to the lowering of operational costs for this algorithm. Finally, with the exception of the last 5 episodes, only E2-D4QN dominates the rest of the trace in the mean reward metric, with the exception of the last five episodes.

### 3.4. Total Scaled Data-Transportation Costs per Episode

The total DT costs per time-step as defined in (13) were computed and the mean values per episode were scaled and plotted in Figure 3d. During the whole evaluation period, both E2-D4QN and GP-LLC incur higher DT costs with respect to every other algorithm. This phenomenon is explained by the Enhanced Exploration mechanism, which permits E2-D4QN and GP-LLC, to accept requests even when the resulting RTT is over the acceptable threshold. E2-D4QN and GP-LLC accept every incoming request instead. Notice, however, that E2-D4QN and GP-LLC progressively reduce DT costs due to common path creation for similar SFCs. From the twentieth episode on, however, only E2-D4QN minimizes such costs while maintaining an acceptance ratio greater than 0.5.

### 3.5. Total Scaled Hosting Costs per Episode

A similar explanation can be given for the total Hosting Cost behavior. Such a cost was computed for each time-step using (11) and the mean values per episode were scaled and plotted in Figure 3e. Given the adaptive resource provisioning algorithm described in Appendix A.1, we can argue that, in general, hosting cost is high for E2-D4QN and GP-LLC because their throughput is high. The hosting costs burst that characterizes the first twenty episodes, however, can be explained by the network initialization state during experiments: Every algorithm is evaluated from an uncommon network state with respect to the steady state reached during training. The algorithms that are equipped with the Enhanced-Exploration mechanism tend to worsen such a performance drop at the beginning of the testing trace because of the unconstrained nature of such mechanism. It is the Enhanced-Exploration however, that drives our proposed agent to learn sub-optimal policies that permit to maximize the network acceptance ratio.

### 3.6. Optimization Objective

The optimization objective as defined in (16) was computed at each time-step, and the mean values per episode were scaled and plotted in Figure 3f. Recall that (16) is invalid whenever the acceptance ratio is above the minimum threshold of 0.5 as mentioned in Section 2.1.4. For this reason, in Figure 3f we set the optimization value to zero whenever the minimum service constraint was not met. Notice that no algorithm can achieve the minimum acceptance ratio during the first ten episodes of the test. This behavior can be explained by the greedy initialization with which every test has been carried out: The initial network state for every algorithm is very different from the states observed during training. From the tenth episode on, however, E2-D4QN is the only algorithm to achieve a satisfactory acceptance ratio, and thus, the optimization objective function has a non-zero value.

## 4. Discussion

Trace-driven simulations have revealed that our approach shows adaptability to the complexity of the particular context of Live-Streaming in vCDN with respect to the state-of-art algorithms designed for general-case SFC deployment. In particular, our simulation test revealed decisive QoS performance improvements in terms of acceptance ratio with respect to any other *backtracking* algorithm. Our agent progressively learns to adapt to the complex environment conditions like different user cluster traffic patterns, different channel popularities, unitary resource provision costs, VNF instantiation times, etc.

We assess the algorithm's performance in a bounded-resource scenario aiming to build a safe-exploration strategy that enables the market entry of new vCDN players. Our experiments have shown that the proposed algorithm, E2-D4QN is the only one to adapt to such conditions, maintaining an acceptance ratio above the general case state-of-art techniques while keeping a delicate balance between network throughput and operational costs.

Based on the results in the previous section, we now argue the main reasons that make E2-D4QN the most suitable algorithm for online optimization of SFC Deployment on a Live-

video delivery vCDN scenario. The main reason for our proposed algorithm's advantage is the combination of the enhanced exploration with a dense-rewards mechanism on a dueling DDQN framework. We argue that such a combination leads to discover convenient long-term actions in contrast to convenient short-term actions during training.

#### 4.1. Environment Complexity Adaptation

As explained in Section 2.3.4, we have compared our E2-D4QN agent with the NFVDeep algorithm presented in [14], with three progressive enhancements to such algorithm, and with an extension of the algorithm presented in [57], which we called GP-LLC. Authors in [14] assumed utilization-independent processing times. A consequence of this assumption is the possibility of computing the remaining latency space with respect to the RTT threshold before each assignment. In our work, instead, we argue that realistic processing times should be modeled as utilization-dependent. Moreover, Xiao et al. did not model ingestion-related utilization nor VNF instantiation time penalties. Relaxing the environment with these assumptions simplifies the environment and helps on-policy DRL schemes like the one in [14] to converge to suitable solutions. Lastly, in NFVDeep, prior knowledge of each SFC session's duration is also assumed. This feature helps the agent to learn to accept longer sessions to increase the throughput. Unfortunately, it is not realistic to assume session duration knowledge when modeling Live-Streaming in vCDN context. Our model is agnostic to this feature and maximizes the overall throughput when optimizing the acceptance ratio. This paper shows that the NFVDeep algorithm cannot reach a good AR on SFC Deployment optimization without assuming all the aforementioned relaxations.

#### 4.2. State Value, Advantage Value and Action Value Learning

In this work, we propose the usage of the dueling-DDQN framework for implementing a DRL agent that optimizes SFC Deployment. Such a framework is meant to learn approximators for the state value function,  $V(s)$ , the action advantage function,  $A(a)$ , and the action-value function,  $Q(s, a)$ . Learning such functions helps to differentiate between relevant actions in the presence of many similar-valued actions. This is the main reason why NFVDeep-D3QN improves AR with respect to NFVDeep: Learning the action advantage function, helps to identify convenient long-term actions from a set of similar valued actions. For example, preserving resources of low-cost nodes for popular channel bursts in the future can be more convenient in the long term with respect to adopting a round-robin load-balancing strategy during low incoming traffic periods. Moreover, suppose we do such a round-robin dispatching. In that case, the SFC routes to content providers won't tend to divide the hosting nodes into non-overlapping clusters. This will provoke more resource usage in the long run: almost every node will ingest the content of almost every content provider. As generally content-ingestion resource usage is much heavier with respect to content-serving, this strategy will accentuate the resource leakage on the vCDN in the long run, provoking bad QoS performance. Our E2-D4QN learns to polarize the SFC routes in order to minimize content ingestion resource usage during the training phase. Such a biased policy performs in the best way possible with respect to the compared algorithms taking into account the whole evaluation period.

#### 4.3. Dense Reward Assignment Policies

Our agent converges to sub-optimal policies by carefully designing a reward schema as the one presented in Section 2.2.2. Our algorithm assigns a specific reward at each MDP transition considering the optimality of VNF assignments in terms of QoS, hosting costs, and data transfer costs. This dense-reward schema enhances the agent's convergence. In fact, in our experiments, we have also noticed that the dense-reward algorithms improve the results of their sparse-reward counterparts. In other words, we see in Figure 3b that NFVDeep-Dense performs slightly better than NFVDeep, and NFVDeep-Dense-D3QN performs better than NFVDeep-D3QN. This improvement exists because dense rewards provide valuable feedback at each assignment step of the SFC, improving convergence

of the DRL agents to shorter RTTs. On the other hand, we have also observed that even if cost-related penalties are sparsely subtracted in our experiments, the proposed DRL agent learns to optimize SFC deployment not only with respect to QoS but also taking into account the operational costs.

#### 4.4. Enhanced Exploration

Notice that GP-LLC does not learn inherent network traffic dynamics, making it impossible to differentiate convenient long-term actions from greedy actions. For example, GP-LLC won't adopt long-term resource consolidation policies, in which each channel is ingested by a defined subset of nodes to amortize the overall ingestion resource consumption. On the other hand, we argue that the main reason that keeps NFVDeep above a satisfactory performance is the leakage of the enhanced exploration mechanism. In fact, we demonstrate that the acceptance ratio of E2-D4QN surpasses every other algorithm because we add enhanced exploration, which enriches the experience replay buffer with non-optimal SFC deployments during training, preventing the agent from getting stuck at local optima, which we think is the main problem with NFVDeep under our environment conditions.

In practice, in all the NFVDeep or backtracking algorithms, i.e., the algorithms without the enhanced-exploration mechanism, at each VNF assignment, the candidate nodes are filtered based on their utilization availability. Only non-overloaded nodes are available candidates. If the agent chooses an overloaded node, the action is ignored, and no reward is given. Our model instead performs assignment decisions without being constrained by current node utilization to enhance the exploration of the action space. The well-designed state-space codifies important features that drive learning towards sub-optimal VNF placements:

1. The request vector codifies useful information about the requests that help our agent extract the incoming traffic patterns.
2. The ingestion vector helps our agent to optimize ingestion-related resource demand by avoiding session assignments to VNF instances that do not ingest the content requested at the moment of assignment.
3. The maximum utilization vector gives our agent resource utilization awareness, making it able to converge to assignment policies that optimize processing times and preserve QoS.

Consequently, our agent learns optimal SFC Deployment policies without knowing the actual bounds of the resource provision or the current instantiation state of the VNF instances that compose the vCDN. It learns to *recognize* the maximum resource provisioning for the VNFs and also learns to evict assignments to non-initialized VNFs thanks to our carefully designed reward assignment scheme.

#### 4.5. Work Limitations and Future Research Directions

We have based the experiments in this paper on a real-world data set concerning a particular video delivery operator. In this case, the hosting nodes of the corresponding proprietary CDN are deployed in the Italian territory. However, such a medium-scale deployment is not the unique possible configuration for a CDN. Consequently, as future work, we plan to obtain or generate data concerning high-scale topologies to assess the scalability of our algorithm to such scenarios.

Further, this paper presents the assessment of the performance of various DRL-based algorithms. However, the authors of this work had access to real-world data set limited to a five-day trace. Consequently, the algorithms presented in this work were trained on a four-day trace, while the evaluation period consisted of a single day. Future research directions include assessing our agent's training and evaluation performance on data concerning more extended periods.

Finally, in this work, we have used a VNF resource-provisioning algorithm that is greedy and reactive, as specified in Appendix A.1. A DRL-based resource-provisioning policy would instead act proactive and long-term convenient actions. Such a resource-

provisioning policy, combined with the SFC deployment policy presented in this work, would further optimize QoS and Costs. Thus, future work also includes the development of a multi-agent DRL framework for the joint optimization of both resource provisioning and SFC deployment tasks in the context of live-streaming in vCDN.

**Author Contributions:** Conceptualization and methodology, J.F.C.M., L.R.C., R.S. and A.S.R.; software, J.F.C.M. and R.S.; investigation, validation and formal analysis J.F.C.M., R.S. and L.R.C.; resources and data curation, J.F.C.M., L.R.C. and R.S.; writing—original draft preparation, J.F.C.M.; writing—review and editing, J.F.C.M., R.S., R.P.C.C., L.R.C., A.S.R. and M.M.; visualization, J.F.C.M.; supervision and project administration, L.R.C., A.S.R. and M.M.; funding acquisition, R.P.C.C. and M.M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by *ELIS Innovation Hub* within a collaboration with *Vodafone* and partly funded by ANID—Millennium Science Initiative Program—NCN17\_129. R.C.C was funded by ANID Fondecyt Postdoctorado 2021 # 3210305.

**Data Availability Statement:** Not applicable, the study does not report any data.

**Acknowledgments:** The authors wish to thank L. Casasús, V. Paduano and F. Kieffer for their valuable insights.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## Abbreviations

The following abbreviations are used in this manuscript:

ANN	Artificial Neural Network
CDN	Content Delivery Network
CP	Content Provider
DT	Data-Transportation
GP-LLC	Greedy Policy of Lowest Latency and Lowest Cost algorithm
ILP	Integer Linear Programming
ISP	Internet Service Provider
QoE	Quality of Experience
QoS	Quality of Service
MANO	Management and orchestration framework
MC	Markov Chain
MDP	Markov Decision Process
MVNO	Mobile Virtual Network Operator
NFV	Network Function Virtualization
NFVI	Network Function Virtualization Infrastructure
OTT	Overt-The-Top Content
RTT	Round-Trip-Time
SDN	Software Defined Networking
SFC	Service Function Chain
vCDN	virtualized-Content Delivery Network
VNF	Virtual Network Function
VNO	Virtual Network Orchestrator

## Appendix A. Further Modelisation Details

### Appendix A.1. Resource Provisioning Algorithm

In this paper we assume that the VNO component is acting a *greedy* resource provisioning algorithm, i.e., the resource provision on  $f_i^k$  for the next time-step will be computed as:

$$c_{res,k,i}^{t+1} = \min(c_{res,k,i}^t \cdot \frac{\mu_{res,k,i}^t}{\hat{\mu}_{res,k,i}}, c_{res,k,i}^{max}), \forall res \in \{cpu, bw, mem\} \quad (A1)$$

where the parameter  $c_{res,k,i}^{max}$  is the maximum *res* resource capacity available for  $f_i^k$ , and  $\hat{\mu}_{res,k,i}$  is a parameter indicating a fixed *desired utilization* of  $f_i^k$  after the adaptation takes place and before receiving further session requests. Recall that  $\mu_{res,k,i}^t$  is the current *res* resource utilization in  $f_i^k$ . Resource adaptation procedure is triggered periodically each  $T_a$  time-steps, where  $T_a$  is a fixed parameter. On the other hand, each time that any  $f_i^k$  is instantiated, the VNO allocates a fixed minimum resource capacity for each resource in such VNF instance, denoted as  $c_{res,k,i}^{min}$ .

Appendix A.2. Inner Delay-Penalty Function

The core of our QoS related reward is the delay-penalty function, which has some properties specified in Section 2.2.1. The function that we used on our experiments is the following:

$$d(t) = \frac{1}{t} + e^{-t} + 2e^{-\frac{t}{100}} + e^{-\frac{t}{500}} - 1 \tag{A2}$$

Notice that the domain of  $d(t)$  will be the RTT of any SFC deployment and the co-domain will be the segment  $[-1, 1]$ . Notice also that:

$$\lim_{t \rightarrow +\infty} d(t) = -1 \text{ and } \lim_{t \rightarrow t_{min}} d(t) \approx 1$$

Such a bounded co-domain helps to stabilize and improve the learning performance of our agent. Notice, however that it is worth noting that similar functions could be easily designed for other values of  $T$ .

Appendix A.3. Simulation Parameters

The whole list of our simulation parameters is presented in Table A1. Every simulation has used such parameters unless other values are explicitly specified.

Table A1. List of simulation parameters.

Parameter	Description	Value
$\gamma_{CPU}$	CPU Unit Resource Costs (URC) (for each cloud provider)	(0.19, 0.6, 0.05)
$\gamma_{MEM}$	Memory URC	(0.48, 1.2, 0.1)
$\gamma_{BW}$	Bandwidth URC	(0.9, 2.5, 0.25)
$c_{max}$	Maximum resource provision parameter (assumed equal for all the resource types)	20
$c_{min}$	Minimum resource provision parameter (assumed equal for all the resource types)	5
$\phi_p$	Payload workload exponent	0.2
$\phi_b$	Bit-rate workload exponent	0.1
$\rho_{cpu}^*$	Optimal CPU Processing Time (baseline of over-usage degradation)	$5 \times 10^{-3}$
$\rho_{mem}^*$	Optimal memory PT	$1 \times 10^{-3}$
$\rho_{bw}^*$	Optimal bandwidth PT	$5 \times 10^{-2}$
$\psi_{cpu}$	CPU exponential degradation base	100
$\psi_{mem}$	Memory deg. b.	100
$\psi_{bw}$	Bandwidth deg. b.	100
$I_{ch}$	cache VNF Instantiation Time Penalization in ms (ITP)	10,000
$I_{st}$	streamer VNF ITP	8000
$I_{co}$	compressor VNF ITP	7000

**Table A1.** Cont.

Parameter	Description	Value
$I_{tr}$	transcoder VNF ITP	11,000
$T_a$	Time-steps per greedy resource adaptation	20
$\hat{\mu}_{res,k,n}$	Desired resulting utilization after adaptation	0.4
$\alpha_{res}$	Optimal resource <i>res</i> utilization (assumed equal for every resource type)	0.75

*Appendix A.4. Training Hyper-Parameters*

A complete list of the hyper-parameters values used in the training cycles is specified in Table A2. Every training procedure has used such values unless other values are explicitly specified.

**Table A2.** List of hyper-parameters’ values for our training cycles.

Hyper-Parameter	Value
Discount factor ( $\gamma$ )	0.99
Learning rate	$1.5 \times 10^{-4}$
Time-steps per episode	80
Initial $\epsilon$ -greedy action probability	0.9
Final $\epsilon$ -greedy action probability	0.0
$\epsilon$ -greedy decay steps	$2 \times 10^5$
Replay memory size	$1 \times 10^5$
Optimization batch size	64
Target-network update frequency	5000

**Appendix B. GP-LLC Algorithm Specification**

In this paper, we have compared our E2-D4QN agent with a greedy policy lowest-latency and lowest-cost (GP-LLC) SFC deployment agent. Algorithm A1 describes the behavior of the GP-LLC agent. Note that the lowest-latency and lowest-cost (LLC) criterion can be seen as a procedure that, given a set of candidate hosting nodes,  $N_H^c$  chooses the correct hosting node to deploy the current VNF request  $\hat{f}_r^k$  of a SFC request  $r$ . Such a procedure is at the core of the GP-LLC algorithm, while the outer part of the algorithm is responsible for choosing the hosting nodes that form the candidate set according to a QoS maximization criterion. The LLC criterion works as follows. Given a set of candidate hosting nodes and a VNF request, the LLC criterion will divide the candidate nodes in subsets considering the cloud provider they come from. It will then choose the hosting node corresponding to the route that will generate less transportation delay, i.e., the fastest route, from the cheapest cloud-provider candidate node subset.

The outer part of the algorithm acts instead as follows. Every time that a VNF request  $\hat{f}_r^k$  needs to be processed, the GP-LLC agent monitors the network conditions. The agent identifies the hosting nodes that are currently not in overload conditions,  $\hat{N}_H$ , the ones that currently have a resource provision that is less than the maximum for all the resource types  $\tilde{N}_H$ , and the hosting nodes that currently have a  $f_k$  ingesting the content from the same content provider requested by  $\hat{f}_r^k$ ,  $N_H^k$  (lines 2–4). Notice that  $\tilde{N}_H$  is the set of nodes whose resource provision can still be augmented by the VNO. Notice also that choosing a node from  $N_H^k$  to assign  $\hat{f}_r^k$ , implies not to incur in a Cache MISS event and consequently warrants the acceptance of  $r$ . If  $\hat{N}_H \cap \tilde{N}_H$  is not an empty set, the agent assigns  $\hat{f}_r^k$  to a node in such a set following the LLC criterion. However, if  $\hat{N}_H \cap \tilde{N}_H$  is an empty set, then if at least  $\hat{N}_H$  is not empty, then a node from  $\hat{N}_H$  will be chosen using the LLC criterion. If on the other hand,  $\tilde{N}_H$  is empty, then a node from  $\tilde{N}_H$  will be chosen with the LLC criterion.

Finally, if both  $\hat{N}_H$  and  $\tilde{N}_H$  are empty sets, then a random hosting node will be chosen for hosting  $f_r^k$  (lines 5–16). Choosing a random node in the last case instead of using the LLC criterion from the whole hosting node will prevent bias in the assignation policy to cheap nodes with fast routes. Making such a random choice will then result in an increment in the overall load balance among the hosting nodes.

Notice that, whenever possible, GP-LLC will evict overloading nodes with assignation actions and will always choose the best actions in terms of QoS. Moreover, given a set of candidate nodes respecting such a greedy QoS-preserving criterion, the inner LLC criterion will tend to optimize hosting costs and data-transmission delays. Notice also that GP-LLC does not take into account data-transportation costs for VNF SFC deployment.

---

**Algorithm A1** GP-LLC VNF Assignation procedure.

---

```

1: for  $f_r^k \in r$  do
2:   Get the non-overloaded hosting nodes set  $\hat{N}_H$ 
3:   Get the still-scalable hosting nodes set  $\tilde{N}_H$ 
4:   Get the set of hosting nodes that currently have a  $f^k$  ingesting  $l_r$  on  $N_l^k$ 
5:   if  $|\hat{N}_H| > 0$  then
6:     if  $|\hat{N}_H \cap \tilde{N}_H| > 0$  then
7:       use the LLC criterion to chose  $f_r^k$  from  $\hat{N}_H \cap \tilde{N}_H$ 
8:     else
9:       use the LLC criterion to chose  $f_r^k$  from  $\hat{N}_H$ 
10:    end if
11:  else
12:    if  $|\tilde{N}_H| > 0$  then
13:      use the LLC criterion to chose  $f_r^k$  from  $\tilde{N}_H$ 
14:    else
15:      choose a random node  $f_r^k$  from  $|\hat{N}_H|$ 
16:    end if
17:  end if
18: end for

```

---

## References

1. Cisco, V. Cisco Visual Networking Index: Forecast and Methodology 2016–2021. *Compleat. Vis. Netw. Index (VNI) Forecast*. **2017**, *12*, 749–759.
2. Budhkar, S.; Tamarapalli, V. An overlay management strategy to improve QoS in CDN-P2P live streaming systems. *Peer-to-Peer Netw. Appl.* **2020**, *13*, 190–206. [CrossRef]
3. Demirci, S.; Sagioglu, S. Optimal placement of virtual network functions in software defined networks: A survey. *J. Netw. Comput. Appl.* **2019**, *147*, 102424. [CrossRef]
4. Li, J.; Lu, Z.; Tong, Y.; Wu, J.; Huang, S.; Qiu, M.; Du, W. A general AI-defined attention network for predicting CDN performance. *Future Gener. Comput. Syst.* **2019**, *100*, 759–769. [CrossRef]
5. Yi, B.; Wang, X.; Li, K.; Das, S.k.; Huang, M. A comprehensive survey of Network Function Virtualization. *Comput. Netw.* **2018**, *133*, 212–262. [CrossRef]
6. Matias, J.; Garay, J.; Toledo, N.; Unzilla, J.; Jacob, E. Toward an SDN-enabled NFV architecture. *IEEE Commun. Mag.* **2015**, *53*, 187–193. [CrossRef]
7. Taleb, T.; Mada, B.; Corici, M.I.; Nakao, A.; Flinck, H. PERMIT: Network slicing for personalized 5G mobile telecommunications. *IEEE Commun. Mag.* **2017**, *55*, 88–93. [CrossRef]
8. Khan, R.; Kumar, P.; Jayakody, D.N.K.; Liyanage, M. A Survey on Security and Privacy of 5G Technologies: Potential Solutions, Recent Advancements, and Future Directions. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 196–248. [CrossRef]
9. Jahromi, N.T. Towards the Softwarization of Content Delivery Networks for Component and Service Provisioning. Ph.D. Thesis, Concordia University, Montreal, QC, Canada, 2018.
10. ETSI. ETSI White Paper No. 24, MEC Deployments in 4G and Evolution towards 5G, 02-2018. Available online: [https://www.etsi.org/images/files/etsiwhitepapers/etsi\\_wp24\\_mec\\_deployment\\_in\\_4g\\_5g\\_final.pdf](https://www.etsi.org/images/files/etsiwhitepapers/etsi_wp24_mec_deployment_in_4g_5g_final.pdf) (accessed on 26 October 2021).
11. Hernandez-Valencia, E.; Izzo, S.; Polonsky, B. How will NFV/SDN transform service provider opex? *IEEE Netw.* **2015**, *29*, 60–67. [CrossRef]
12. Cziva, R.; Pezaros, D.P. Container Network Functions: Bringing NFV to the Network Edge. *IEEE Commun. Mag.* **2017**, *55*, 24–31. [CrossRef]

13. Herbaut, N.; Negru, D.; Chen, Y.; Frangoudis, P.A.; Ksentini, A. Content Delivery Networks as a Virtual Network Function: A Win-Win ISP-CDN Collaboration. In Proceedings of the 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, USA, 4–8 December 2016.
14. Xiao, Y.; Zhang, Q.; Liu, F.; Wang, J.; Zhao, M.; Zhang, Z.; Zhang, J. NFVdeep: Adaptive online service function chain deployment with deep reinforcement learning. In Proceedings of the International Symposium on Quality of Service; Number Article 21 in IWQoS '19; Association for Computing Machinery: New York, NY, USA, 2019; pp. 1–10.
15. Lukovszki, T.; Schmid, S. Online Admission Control and Embedding of Service Chains. In *Structural Information and Communication Complexity*; Springer International Publishing: Berlin/Heidelberg, Germany, 2015; pp. 104–118.
16. Huang, W.; Zhu, H.; Qian, Z. AutoVNF: An Automatic Resource Sharing Schema for VNF Requests. *J. Internet Serv. Inf. Secur.* **2017**, *7*, 34–47.
17. Wowza Media Systems: 4 Tips for Sizing Streaming Server Hardware. 2017. Available online: <https://www.wowza.com/blog/4-tips-for-sizing-streaming-server-hardware> (accessed on 30 September 2021).
18. Etsi, G. 001, Network Functions Virtualization (NFV): Use Cases, October 2013. Available online: [https://www.etsi.org/deliver/etsi\\_gs/nfv/001\\_099/001/01.01.01\\_60/gs\\_nfv001v010101p.pdf](https://www.etsi.org/deliver/etsi_gs/nfv/001_099/001/01.01.01_60/gs_nfv001v010101p.pdf) (accessed on 26 October 2021).
19. Herbaut, N. Collaborative Content Distribution over a VNF-as-a-Service Platform. Ph.D. Thesis, Université de Bordeaux, Bordeaux, France, 2017.
20. Benkacem, I.; Taleb, T.; Bagaa, M.; Flinck, H. Optimal VNFs Placement in CDN Slicing Over Multi-Cloud Environment. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 616–627. [[CrossRef](#)]
21. Gil Herrera, J.; Botero, J.F. Resource Allocation in NFV: A Comprehensive Survey. *IEEE Trans. Netw. Serv. Manag.* **2016**, *13*, 518–532. [[CrossRef](#)]
22. Fei, X.; Liu, F.; Xu, H.; Jin, H. Adaptive VNF Scaling and Flow Routing with Proactive Demand Prediction. In Proceedings of the IEEE INFOCOM 2018—IEEE Conference on Computer Communications, Honolulu, HI, USA, 16–19 April 2018; pp. 486–494.
23. Zhang, R.X.; Ma, M.; Huang, T.; Pang, H.; Yao, X.; Wu, C.; Liu, J.; Sun, L. Livesmart: A QoS-Guaranteed Cost-Minimum Framework of Viewer Scheduling for Crowdsourced Live Streaming. In Proceedings of the 27th ACM International Conference on Multimedia; MM '19; Association for Computing Machinery: New York, NY, USA, 2019; pp. 420–428.
24. Gupta, L.; Jain, R.; Erbad, A.; Bhamare, D. The P-ART framework for placement of virtual network services in a multi-cloud environment. *Comput. Commun.* **2019**, *139*, 103–122. [[CrossRef](#)]
25. Tomassilli, A.; Giroire, F.; Huin, N.; Pérennes, S. Provably Efficient Algorithms for Placement of Service Function Chains with Ordering Constraints. In Proceedings of the IEEE INFOCOM 2018—IEEE Conference on Computer Communications, Honolulu, HI, USA, 16–19 April 2018; pp. 774–782.
26. Ibn-Khedher, H.; Abd-Elrahman, E.; Kamal, A.E.; Afifi, H. OPAC: An optimal placement algorithm for virtual CDN. *Comput. Netw.* **2017**, *120*, 12–27. [[CrossRef](#)]
27. Ibn-Khedher, H.; Hadji, M.; Abd-Elrahman, E.; Afifi, H.; Kamal, A.E. Scalable and Cost Efficient Algorithms for Virtual CDN Migration. In Proceedings of the 2016 IEEE 41st Conference on Local Computer Networks (LCN), Dubai, United Arab Emirates, 7–10 November 2016; pp. 112–120.
28. Yala, L.; Frangoudis, P.A.; Lucarelli, G.; Ksentini, A. Cost and Availability Aware Resource Allocation and Virtual Function Placement for CDNaas Provision. *IEEE Trans. Netw. Serv. Manag.* **2018**, *15*, 1334–1348. [[CrossRef](#)]
29. Filelis-Papadopoulos, C.K.; Endo, P.T.; Bendechache, M.; Svorobej, S.; Giannoutakis, K.M.; Gravvanis, G.A.; Tzovaras, D.; Byrne, J.; Lynn, T. Towards simulation and optimization of cache placement on large virtual content distribution networks. *J. Comput. Sci.* **2020**, *39*, 101052. [[CrossRef](#)]
30. Dieye, M.; Ahvar, S.; Sahoo, J.; Ahvar, E.; Glitho, R.; Elbiaze, H.; Crespi, N. CPVNF: Cost-Efficient Proactive VNF Placement and Chaining for Value-Added Services in Content Delivery Networks. *IEEE Trans. Netw. Serv. Manag.* **2018**, *15*, 774–786. [[CrossRef](#)]
31. Jahromi, N.T.; Kianpisheh, S.; Glitho, R.H. Online VNF Placement and Chaining for Value-added Services in Content Delivery Networks. In Proceedings of the 2018 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), Washington, DC, USA, 25–27 June 2018; pp. 19–24.
32. Jia, Y.; Wu, C.; Li, Z.; Le, F.; Liu, A. Online Scaling of NFV Service Chains Across Geo-Distributed Datacenters. *IEEE/ACM Trans. Netw.* **2018**, *26*, 699–710. [[CrossRef](#)]
33. Das, B.C.; Takahashi, S.; Oki, E.; Muramatsu, M. Approach to problem of minimizing network power consumption based on robust optimization. *Int. J. Commun. Syst.* **2019**, *32*, e3891. [[CrossRef](#)]
34. Xie, Y.; Wang, B.; Wang, S.; Luo, L. Virtualized Network Function Provisioning in Stochastic Cloud Environment. In Proceedings of the ICC 2020—2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020, pp. 1–6.
35. Marotta, A.; Kassler, A. A Power Efficient and Robust Virtual Network Functions Placement Problem. In Proceedings of the 2016 28th International Teletraffic Congress (ITC 28), Würzburg, Germany, 12–16 September 2016.
36. Marotta, A.; Zola, E.; D'Andreagiovanni, F.; Kassler, A. A fast robust optimization-based heuristic for the deployment of green virtual network functions. *J. Netw. Comput. Appl.* **2017**, *95*, 42–53. [[CrossRef](#)]
37. Ito, M.; He, F.; Oki, E. Robust Optimization Model for Probabilistic Protection under Uncertain Virtual Machine Capacity in Cloud. In Proceedings of the 2020 16th International Conference on the Design of Reliable Communication Networks DRCN 2020, Milan, Italy, 25–27 March 2020.

38. Mijumbi, R.; Gorricho, J.; Serrat, J.; Claeys, M.; De Turck, F.; Latré, S. Design and evaluation of learning algorithms for dynamic resource management in virtual networks. In Proceedings of the 2014 IEEE Network Operations and Management Symposium (NOMS), Krakow, Poland, 5–9 May 2014; pp. 1–9.
39. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv* **2015**, arXiv:cs.LG/1509.02971.
40. Dulac-Arnold, G.; Evans, R.; van Hasselt, H.; Sunehag, P.; Lillicrap, T.; Hunt, J.; Mann, T.; Weber, T.; Degris, T.; Coppin, B. Deep Reinforcement Learning in Large Discrete Action Spaces. *arXiv* **2015**, arXiv:cs.AI/1512.07679.
41. Wang, Z.; Schaul, T.; Hessel, M.; Hasselt, H.; Lanctot, M.; Freitas, N. Dueling Network Architectures for Deep Reinforcement Learning. In *Proceedings of Machine Learning Research*; PMLR: New York, NY, USA, 2016; Volume 48, pp. 1995–2003.
42. Reis, J.; Rocha, M.; Phan, T.K.; Griffin, D.; Le, F.; Rio, M. Deep Neural Networks for Network Routing. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8.
43. Quang, P.T.A.; Hadjadj-Aoul, Y.; Outtagarts, A. A Deep Reinforcement Learning Approach for VNF Forwarding Graph Embedding. *IEEE Trans. Serv. Manag.* **2019**, *16*, 1318–1331. [[CrossRef](#)]
44. Anh Quang, P.T.; Hadjadj-Aoul, Y.; Outtagarts, A. Evolutionary Actor-Multi-Critic Model for VNF-FG Embedding. In Proceedings of the 2020 IEEE 17th Annual Consumer Communications Networking Conference (CCNC), Las Vegas, NV, USA, 10–13 January 2020; pp. 1–6.
45. Yan, Z.; Ge, J.; Wu, Y.; Li, L.; Li, T. Automatic Virtual Network Embedding: A Deep Reinforcement Learning Approach With Graph Convolutional Networks. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 1040–1057. [[CrossRef](#)]
46. Khezri, H.R.; Moghadam, P.A.; Farshbafan, M.K.; Shah-Mansouri, V.; Kebriaei, H.; Niyato, D. Deep Reinforcement Learning for Dynamic Reliability Aware NFV-Based Service Provisioning. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6.
47. Mao, W.; Wang, L.; Zhao, J.; Xu, Y. Online Fault-tolerant VNF Chain Placement: A Deep Reinforcement Learning Approach. In Proceedings of the 2020 IFIP Networking Conference (Networking), Paris, France, 22–26 June 2020; pp. 163–171.
48. Pei, J.; Hong, P.; Pan, M.; Liu, J.; Zhou, J. Optimal VNF Placement via Deep Reinforcement Learning in SDN/NFV-Enabled Networks. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 263–278. [[CrossRef](#)]
49. Santos, G.L.; Kelner, J.; Sadok, D.; Endo, P.T. Using Reinforcement Learning to Allocate and Manage SFC in Cellular Networks. In Proceedings of the 2020 16th International Conference on Network and Service Management (CNSM), Izmir, Turkey, 2–6 November 2020; pp. 1–5.
50. Beck, M.T.; Botero, J.F. Coordinated Allocation of Service Function Chains. In Proceedings of the 2015 IEEE Global Communications Conference (GLOBECOM), San Diego, CA, USA, 6–10 December 2015; pp. 1–6.
51. Pei, J.; Hong, P.; Xue, K.; Li, D. Resource Aware Routing for Service Function Chains in SDN and NFV-Enabled Network. *IEEE Trans. Serv. Comput.* **2018**, *14*, 985–997. [[CrossRef](#)]
52. Pei, J.; Hong, P.; Xue, K.; Li, D. Efficiently Embedding Service Function Chains with Dynamic Virtual Network Function Placement in Geo-Distributed Cloud System. *IEEE Trans. Parallel Distrib. Syst.* **2019**, *30*, 2179–2192. [[CrossRef](#)]
53. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; Riedmiller, M. Playing Atari with Deep Reinforcement Learning. *arXiv* **2013**, arXiv:cs.LG/1312.5602.
54. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [[CrossRef](#)]
55. van Hasselt, H.; Guez, A.; Silver, D. Deep Reinforcement Learning with Double Q-learning. *arXiv* **2015**, arXiv:cs.LG/1509.06461.
56. Hasselt, H. Double Q-learning. *Adv. Neural Inf. Process. Syst.* **2010**, *23*, 2613–2621.
57. Sun, C.; Bi, J.; Zheng, Z.; Hu, H. SLA-NFV: An SLA-aware High Performance Framework for Network Function Virtualization. In *Proceedings of the 2016 ACM SIGCOMM Conference*; SIGCOMM '16; Association for Computing Machinery: New York, NY, USA, 2016; pp. 581–582.



Review

# Self-Organizing Networks for 5G and Beyond: A View from the Top

Andreas G. Papidas \* and George C. Polyzos

Mobile Multimedia Laboratory, Department of Informatics, School of Information Sciences and Technology, Athens University of Economics and Business, 10434 Athens, Greece; polyzos@aueb.gr

\* Correspondence: papidas@aueb.gr

**Abstract:** We describe self-organizing network (SON) concepts and architectures and their potential to play a central role in 5G deployment and next-generation networks. Our focus is on the basic SON use case applied to radio access networks (RAN), which is self-optimization. We analyze SON applications' rationale and operation, the design and dimensioning of SON systems, possible deficiencies and conflicts that occur through the parallel operation of functions, and describe the strong reliance on machine learning (ML) and artificial intelligence (AI). Moreover, we present and comment on very recent proposals for SON deployment in 5G networks. Typical examples include the binding of SON systems with techniques such as Network Function Virtualization (NFV), Cloud RAN (C-RAN), Ultra-Reliable Low Latency Communications (URLLC), massive Machine-Type Communication (mMTC) for IoT, and automated backhauling, which lead the way towards the adoption of SON techniques in Beyond 5G (B5G) networks.

**Keywords:** self-organization; self-optimization; self-healing; SON applications and architecture; SON design and dimensioning; machine learning (ML) and artificial intelligence (AI) for SON; massive Machine-Type Communication (mMTC); IoT; URLLC; backhauling; SON for 3G/4G; 5G and B5G networks

**Citation:** Papidas, A.G.; Polyzos, G.C. Self-Organizing Networks for 5G and Beyond: A View from the Top. *Future Internet* **2022**, *14*, 95. <https://doi.org/10.3390/fi14030095>

Academic Editor: Michael Mackay

Received: 13 February 2022

Accepted: 14 March 2022

Published: 17 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Almost all industries shall be digitally transformed and accelerated through the launch of 5G networks, which have already expanded dynamically, the COVID-19 pandemic notwithstanding. The development of 3G (Universal Mobile Telecommunications System (UMTS) and High-Speed Packet Access (HSPA)) networks has been focused on advancements at the physical layer of the radio interface and led to higher capacities, while 4G (Long-Term Evolution (LTE), and Long-Term Evolution Advanced (LTE-A)) networks have provided a new IP-based core network architecture on top of extra efficient radio transmission schemes. Fifth-generation networks aim to extend the existing capabilities of 4G (LTE) networks concurrently in the core and access domains via new techniques, but also through pushing digitalization, automation, and interdependence in many, if not all, vertical domains, industries, and aspects of life.

The expected outcome of 5G development and the basic 5G use cases is the provision of extreme mobile broadband (e-MBB) services, enabling a very high data rate, massive Machine-Type Communication (mMTC) for Internet of Things (IoT), the connectivity of a vast number of low-complexity and low-energy-consumption IoT devices capable of monitoring infrastructure, environmental parameters, and logistics, and, finally, Ultra-Reliable Low Latency Communication (URLLC) services supporting applications with strict and very low latency and reliability requirements, which is perhaps the most tricky part [1–8].

As far as the migration from existing Global System for Mobile communication (GSM)/UMTS and LTE networks to 5G is concerned, key enablers include techniques

such as Network Function Virtualization (NFV), aiming to decouple the dependency between hardware and software in terms of network functions and migrate legacy network functions to a virtual or Cloud-based architecture in the core network domain. NFV combined with Software Defined Networking (SDN), with the aim to separate the data and control planes, shall be applied in the radio access network (RAN) part as well through the Cloud-RAN (C-RAN) concept, aiming to virtualize and aggregate the baseband units of the base stations in a central baseband pool [1–8].

Apart from NFV, SDN, C-RAN and techniques applied in the radio access domain, such as massive Multiple Input Multiple Output (MIMO), mmWave technologies, spectrum sharing, network slicing and cognitive radio, the key solution for automatically configuring, optimizing and tuning 5G and B5G network functions is using self-organizing network (SON) platforms, combined with the previously mentioned applications and empowered by intelligent machine learning (ML) algorithms, leading to full RAN automation, self-configuration, self-healing, and self-optimization capabilities [8–23].

The application of self-organization in mobile networks is derived from natural or biological sciences, mainly biology, and a big variety of papers exists in the literature referring to this specific issue; however, we shall not focus on describing the abstract characteristics of self-organization in nature, but describe and analyze the characteristics of SON technology, its benefits, its operation rationale for existing mobile networks, its evolution and the basic application fields in cellular communications networks [24].

### *1.1. Motivation and Main Drivers for SON Deployment in Next-Generation Networks*

The SON concept applied in networking and the telecommunications industry resulted from a long history of original academic research, through industry–academia consortia and standardization organizations, resulting in network automation techniques and leading operators and equipment manufacturers to apply it as the basic solution for automating network functions and optimization processes, especially in the radio access domain. SON platforms are already being used with great success, leading to network performance and Key Performance Indicator (KPI) improvements as well as OPEX/CAPEX benefits in 2G/3G/4G radio access networks of mobile operator networks; interest has been increasing significantly in the last few years as 5G network deployments gradually expand and operators start switching off 2G and 3G networks in order to migrate the frequency usage to 4G and 5G network structures [9–23,25].

The basic trigger point for the development of SON systems is closely related to the automation benefits of ML and artificial intelligence (AI) techniques in order to overcome the manual handling of very complex and real-time aspects of network management, planning, and optimization procedures. As an example, consider the existing scenario for real mobile infrastructures where an operator intends to expand or modernize an existing multi-technology network either in dense urban or rural area cell clusters. During this long process, new base stations (macro, micro, pico, and femto cells and repeaters) enter or leave the network and relocate, and daily issues, such as low-quality channels or interference, must be mitigated manually. Prior to SON, the first step in this process was to manually configure the basic RAN, transport, and core network parameters, while at the same time redesigning the neighboring area network. If we include as a factor the unpredictable user mobility and activity, it is clear that the manual handling of all these procedures has very high complexity and is time consuming and error prone [26].

These challenges can be addressed by SON platforms/systems since network parameter tuning and optimization can be performed automatically through intelligent SON algorithms in real time. This is achieved through decisions made by the SON platform algorithm, which takes real-time network data as input (e.g., network counters collected by various network entities, such as the radio access network controllers—Base Station Controller (BSC), Radio Network Controller (RNC), Mobility Management Entity (MME), and Access and Mobility Management Function (AMF), for 2G, 3G, 4G, and 5G, respectively—at various granularities).

It can be said that as networks evolve following the network deployment lifecycle with procedures starting with their initial planning, network design, installation and commissioning, optimization and finally maintenance, only some sub-procedures such as site survey, selection, licensing, equipment shipment and installation or hardware replacement cannot be substituted by SON platforms [16,27].

Additionally, especially considering the RAN optimization aspect, which is our focus, the following figure depicts the comparison of a traditional, legacy (manual) approach compared to automated optimization through SON. The benefits, as shown in Figure 1, are important since the conventional radio access network optimization process takes approximately 8 weeks, which is slower than the automated approach. More specifically, the conventional optimization rationale includes three basic steps that have to be repeated manually many times until the optimum result is achieved. In order to optimize a RAN, the first step is to collect data from drive tests and measurement campaigns, KPIs, customer complaints and existing configuration. Step 1 is the input to step 2 that includes the reconfiguration and optimization steps so that network performance is improved. Finally, step 3 includes the monitoring of the changes and reconfiguration deployed during step 2 and it provides feedback about how beneficial these are. The process repeats until the optimum result is achieved in a manual manner, leading to a long and time-consuming process. Network automation is the key to eliminating delays and the error-prone nature of manual activities by replacing the prementioned steps with automated procedures performed by ML algorithms [9,28,29].

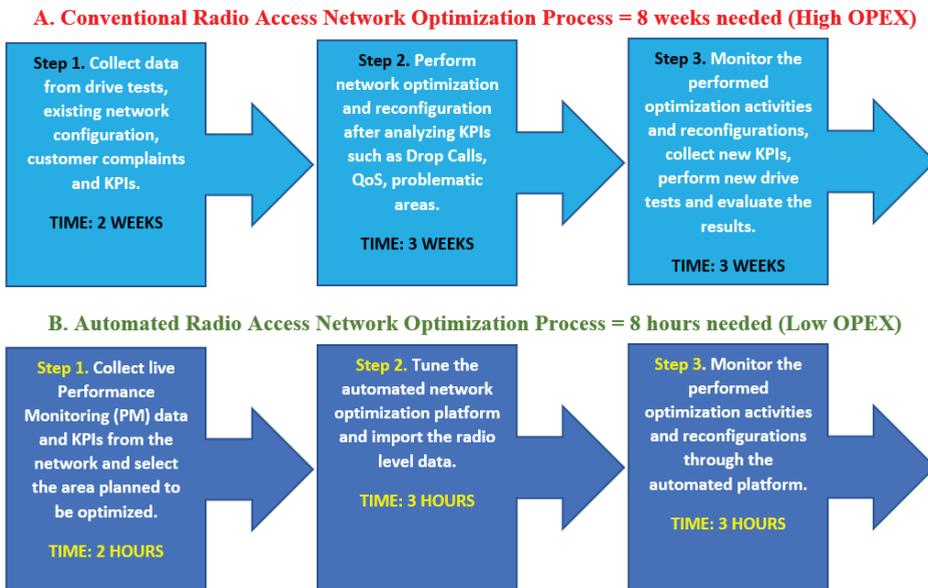


Figure 1. Traditional vs. automated optimization.

Key drivers for SON deployment in existing and next generation networks:

- B5G networks further increase the complexity of network planning and optimization processes since the former must coexist and counteract with the existing networks (2G/3G/4G/5G) in parallel. The network parameter design, tuning, and management lead to a high complexity in terms of smooth coordination among all existing networks, and it is obvious that as more complex architectures and B5G evolve, network management shall not be feasible without the use of automated SON functions operating in real time and taking direct feedback from the network [8–23].

- The standardization of network traffic characteristics and parameters in order to support the basic 5G use cases (i.e., e-MBB, URLLC and MTC) should be embedded in SON systems and applications so that already set targets for 5G, such as low latency and high reliability, can be achieved.
- The benefits of AI and ML for the operation of next-generation networks should be applied as the basic building blocks for B5G intelligent operation. SON platform advancements and network automation are clearly dependent on advanced ML algorithms and their implementations [8–23,30]. Simplistic automation solutions do not enable a distributed approach and they are not effective and efficient in complex environments.
- Academia and the telecommunications industry to combine 5G use cases and intelligent techniques, such as C-RAN, mmWave, cognitive radio (CR), network slicing, spectrum sharing, NFV/SDN, mm-Wave (Millimeter-Wave), automated backhauling and massive MIMO (massive multiple-input and multiple-output). with next-generation SON platforms for maximum performance.

### 1.2. Paper Structure and Contribution

The goal of this paper is to provide an extensive review of self-organizing network (SON) technologies, explain their rationale and operation with a focus on self-optimization functions, describe in detail the flow of SON systems design, analyze basic issues that should be resolved as we move towards SON adoption in 5G networks and beyond, and discuss the latest research directions in this field. In Section 2, we describe the basic SON architectures and provide a detailed guide to the design and dimensioning of SON systems. In Section 3, we analyze the operation of basic self-optimizing applications. We provide a brief description of ML algorithms applied to SON systems in Section 4, and in Section 5 we discuss and comment on the latest research directions for the application of SON technology in different domains of B5G networks. Finally, in Section 6, we provide a summary and conclude the paper.

## 2. SON Operation Rationale, Use Cases, Standardization, Architectures and Dimensioning

### 2.1. Manual RAN Planning and Optimization Activities Replaced by SON

As already mentioned, SON applications were developed to provide network automation and fully replace the RAN optimization process as a main aim and the planning process as a secondary aim, in terms of initial RAN nodes and core/backhaul network configuration, since some steps of network planning such as site location selection and coverage planning cannot be fully automated. The basic steps of the planning lifecycle are depicted in Figure 2 [22,23,28].

The following figure describes the planning process from the early stage of detecting the need for a new site to the activation of an LTE site. As a very first step, the need of a new site is recognized based on known coverage holes or the capacity advancement needs for specific areas. After the radio propagation environment is examined and simulated and exact capacity needs are identified, a coverage simulation is deployed and the initial RAN parameters are designed by taking into account any neighboring sites. A configuration file is created and planned to be committed by the OSS to the RAN, and then the activation takes place. As a last step, real-time network KPIs are monitored and drive testing takes place so that the whole coverage area is examined and a clear outcome about any possible optimization actions needed is extracted. Apart from the initial steps of understanding the need for a new site due to poor coverage or capacity needs and coverage simulation, the configuration procedure can be substituted by SON self-configuration applications that are able to eliminate human errors since the interaction with neighboring sites is what creates complexity.

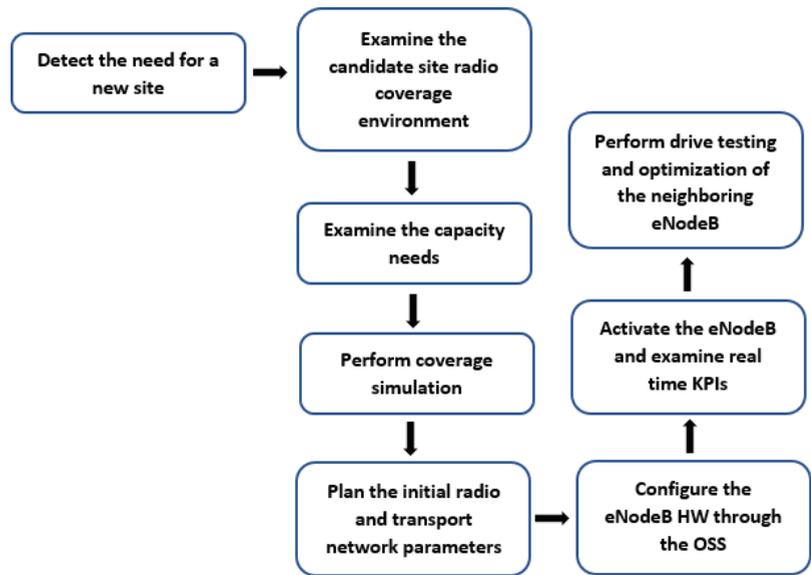


Figure 2. RAN planning and configuration lifecycle.

SON system intervention, no matter the technology (2G/3G/4G/5G), starts from the configuration phase (preoperational state of a node/cell) and expands to network optimization, which is the basic field of SON applications as well as healing. Optimization and healing take place in the operational state, while both preoperational and operational states can migrate from a manual to fully automated status, as depicted in Figure 3, through a SON orchestrator that monitors and organizes SON applications.

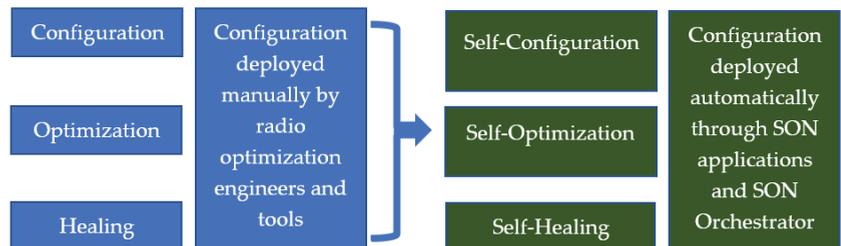


Figure 3. Replacing manual operations with SON.

Healing activities regard the automated maintenance handling in case nodes fail and existing SON systems follow a reactive approach; however, a proactive approach is more efficient. Typical healing cases that lead to a reduction in operational expenses include interference mitigation, the detection of KPI degradation, outage detection, diagnosis and compensation [31].

## 2.2. SON EU Research Programs, the NGMN Alliance and 3GPP Standards

Since SON seem to be the optimum automation technique for mobile networks, academy, industry, alliance and standardization bodies started analyzing potential scenarios and use cases.

Organizations such as 5GPPP and European research programs such as SOCRATES, SEMAFOUR, SESAME, COGNET, Gandalf, and BeFEMTO describe and propose SON architectures [32–39], while the NGMN alliance, founded by leading international mobile

network operators in 2006 as an industry organization to provide innovation platforms for next-generation mobile networks, was the first to propose SON use cases and prerequisites for deployment in modern mobile networks [27,40,41].

During 2017, NGMN proceeded with the analysis of SON use cases for 5G networks since industry adopted SON solutions for 2G/3G/4G networks with great success in terms of network KPI improvement and CAPEX/OPEX savings [21,42–56]. Based on the former NGMN recommendations for SON applicability in 2G/3G/4G networks, NGMN proceeded by proposing SON functions for 5G networks [20].

Just after NGMN, 3GPP started working towards standardizing self-optimizing and self-organizing network use cases for 3G/4G networks, starting from Release 8 up to Release 12. The specific standards are described in detail in network automation and management features. 3GPP releases added enhancements in the prementioned configuration and optimization processes, while proposals allowing inter-radio access technology operation, enhanced inter-cell interference coordination, coverage and capacity optimization, energy efficiency, and the minimization of operational expenses through the minimization of drive tests were added [57–63].

As expected, 3GPP has been working on 5G SON recommendations by further expanding the specific use cases proven to be successful in 2G/3G/4G networks [8,21,64].

### 2.3. Essential SON Use-Cases

According to 3GPP and NGMN, SON use cases are classified into three basic categories with each category including subcategories of SON applications. The three basic ones and key pillars are:

- Self-optimization.
- Self-configuration.
- Self-healing.

Self-optimization includes activities such as coverage and capacity optimization, interference handling, Mobility Load Balancing and cell neighbor relation setting; self-healing relates mainly to maintenance issues such as cell outage detection; and self-configuration enables auto-connectivity and automated initial parameter configuration [23,27,31].

Especially for the cases of self-optimization and self-configuration, it should be noted that the specific procedures introduced in 3GPP TS 36.300 version 9.4.0 Release 9, as a capability of Evolved Universal Terrestrial Radio Access Network (E-UTRAN), were further developed up to their applicability in 5G [58,64].

### 2.4. SON Systems Basic Architectures and Operation Rationale

In terms of SON system architectures, the 3GPP consortium defines the following three types of SON architectures that can be applied in 2G/3G/4G and 5G networks [57–64]:

- Centralized (C-SON).
- Distributed (D-SON).
- Hybrid (H-SON).

In centralized SON (C-SON) architectures, a SON platform that resides in a server cluster handles all network elements centrally through the interaction with the OSS (Operations Support System) of each RAN technology. C-SON is the most widespread solution for mobile operators; however, it might present single point of failure characteristics. On the other hand, the basic advantage is that the overall network status information/KPIs are collected and processed centrally, and thus the SON platform has a better insight regarding the network health status. In Distributed SON (D-SON) cases, SON functions run locally at the base stations, while Hybrid SON (H-SON) is a combination of the prementioned cases that is not widely used.

Moreover, in terms of infrastructure, SON platforms reside on server clusters which host the SON applications running per radio access technology (GSM/UMTS/LTE/5G), the database of the system, the orchestrator, which is the entity managing the applications,

and the API (Application Programming Interface). The system interacts directly and in real time with all basic radio access network controllers (BSC/RNC/MME/AMF) and the OSS and collects live performance counters and KPIs, which are analyzed by the running algorithm of each SON application. The outcome is the automated creation of an executable script executed with no manual intervention through the OSS. More specifically, each script is executed through a dedicated privileged SON OSS account and then an evaluation period/timeframe starts where the SON application algorithm decides to revert the action/script created and executed [27]. A high-level LTE SON system design is proposed in Figure 4 and since 5G SON systems derive from LTE, the same design rationale applies to 5G as well and the only difference is that MME is replaced by AMF [13]. A SON platform consists of the hardware part, including the servers and RAN database, the management of SON applications and orchestration module, which handles SON applications and APIs, providing the ability to handle and manage the RAN database. A SON server cluster interacts with the LTE OSS system of each RAN vendor and uses the latter to commit configuration plans that include the changes that each SON application decides. Configuration plans are committed to the OSS of each vendor in case more than one exists and through the MME to the base stations.

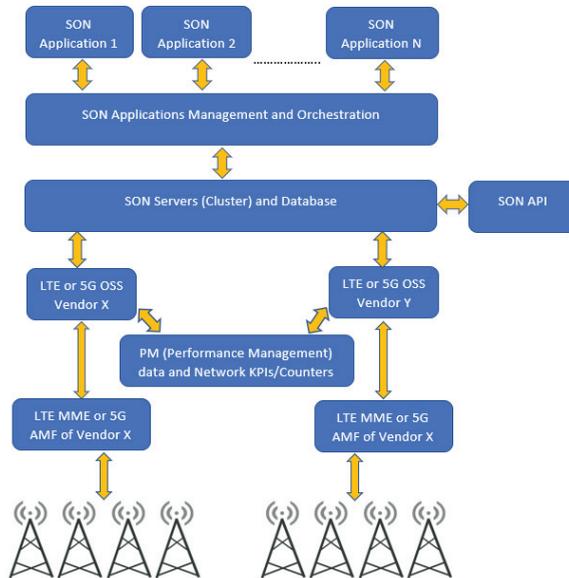


Figure 4. High-level SON system architecture.

As far as the applications are concerned, the rationale that rules all applications is divided into three phases. As depicted in Figure 5, the snapshot period concerns the evaluation of the current state of the network according to the counters of KPI collection prior to any SON actions taken, the action phase concerns the creation and execution of a script for the OSS through SON applications, and the feedback period concerns the evaluation of the action phase and a decision to keep a specific action in the network or revert it to the initial pre-SON operation status [55,65].

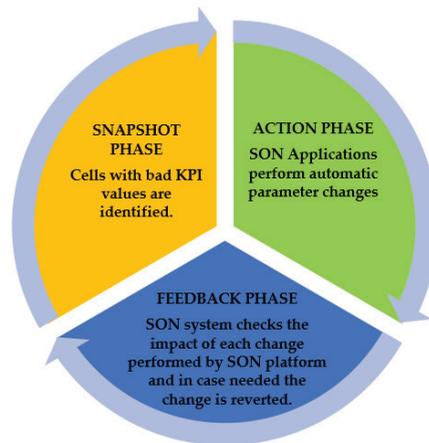


Figure 5. SON loop, basic phases, and lifecycle.

### 2.5. Principles of Dimensioning and Designing a SON System

SON systems' initial dimensioning is based on the collection of existing RAN characteristics through a RAN audit providing key information about the existing policies followed, while it is the most important activity and the basic factor to decide the system architecture and the applications that should be applied. More specifically, in order to decide on the system architecture, we must receive an initial input and define the basic needs no matter what the network technology is that we want to apply SON to (2G/3G/4G/5G). Essential information that must be collected from the network for the needs of system dimensioning and architecture design can be found in the following:

1. Network features applied in the RAN per vendor.
2. RAN parameter settings/values. Key points include:
  - Handover and reselection policy.
  - Admission control policy (admission control is the process that evaluates the existing resources to check if these are sufficient, prior to the establishment of a new connection).
  - Load balancing policy (load balancing aims to transfer the load from overloaded cells to the neighboring less loaded ones so that end-user experience and network performance is improved).
  - Scrambling codes planning strategy in UMTS and Physical Cell ID (PCI) planning strategy in LTE and 5G.
  - Neighbor relation planning strategy (unidirectional/bidirectional) for Intra-Frequency/Inter-Frequency and Inter-RAT and number of maximum neighbors per cell for all technologies.
  - Number of maximum number of cells supported per BSC/RNC/MME/AMF.
  - Inter-carrier Layer Management Strategy (LMS).
  - Frequency bands and number of carriers used per technology (UMTS 2100 MHz, LTE 900 MHz, 1800 MHz, 2600 MHz, etc., according to the spectrum usage acquisition and spectrum refarming policy for 5G).
3. Radio Frequency (RF) hardware equipment in the network (antenna types, Remote Electrical Tilt (RET) types).
4. Pre-SON network KPIs and the expected result after SON deployment.
5. Performance Monitoring (PM) counters activated per vendor.
6. Desired IP planning information for SON platform by considering all needed inter-connections and ports that must be open.

7. Current and expected OSS dimensioning information. (Moreover, full read/write execution rights to the OSS assigned to the SON orchestrator should be ensured.)
8. The geolocation information (latitude and longitude) of the cells as well as antenna azimuth information.
9. Possible network expansion and site rehomeing plans from a BSC/RNC/MME/AMF to a neighboring one, or the use of new RAN technologies.
10. Network size in terms of the number of subscribers and number of active cells per technology, number of network controllers (BSCs, RNCs, MMEs, and AMFs) and number of OSS systems per vendor. A typical real case scenario for a medium-sized operator in Europe prior to 5G commercial launch might include 6,000,000 subscribers in total, 60,000 GSM cells, 90,000 UMTS cells, 80,000 LTE cells, 25 BSCs and RNCs and 5 MMEs. The number of OSS subsystems depends on the different vendors an operator might use.

Table 1 can be used as input to the SON dimensioning and design process for a real network including 2G/3G and LTE technologies and correlates the applied handover, admission control, and load balancing policies with specific SON applications. The same rationale applies in 5G networks as well, leading to the outcome that IRAT policies, admission control and load balancing parameters are the initial factors to consider.

**Table 1.** Correlation of basic radio access parameters with SON applications.

LTE to 3G layer IRAT handover rules	SON-related application according to 3GPP defined use cases
Circuit Switching (CS) fallback to 2nd carrier	ANR (Automatic Neighbor Relations)
Packet Switching (PS) redirection to 2nd carrier	
Cell reselection to 1st carrier	
GSM to 3G layer IRAT handover rules	SON-related application according to 3GPP defined use cases
GSM to UMTS: Cell reselection only (no handover)	ANR (Automatic Neighbor Relations)
UMTS to GSM: Cell reselection and no handover	
Define the value of admission control parameters (UMTS)	SON-related application according to 3GPP defined use cases
Power admission control settings: 1st carrier: 85%; 2nd and 3rd carriers: 75%	MLB (Mobility Load Balancing)
Codes admission control: 90%	
UL/DL channel elements admission control: 95%	
Describe the load balancing features activated	SON-related application according to 3GPP defined use cases
Example: Intrafrequency load sharing	MLB (Mobility Load Balancing)

Through using the RAN audit and information collection as the dimensioning input, the equivalent output regards the optimum architecture design of a 2G/3G/4G/5G SON system. Basic system design decisions include:

1. Decision regarding the hardware infrastructure needs of the SON system to be deployed and prediction for possible hardware expansion (mainly based on the input collected regarding the number of cells per technology as well as existing network controllers (BSC/RNC/MME/AMF).
2. Decision about the needed SON applications and tuning of the latter according to the current network planning and optimization strategy.
3. Decision and prediction about the expected KPI improvement.

Technoeconomic decisions such as system pricing, human resources and system deployment project time plans are taken as well but these are out of the scope of this paper.

In terms of hardware infrastructure, the prementioned input shall lead to the decision of the servers' cluster (number/type of servers) hosting the platform and applica-

tions and providing disaster recovery, database replication, system orchestration and load balancing capabilities.

We suggest that SON platforms interact with a geolocation database including all needed cell antenna directions, antenna type, cell name, site ID, operating frequency, BSC/RNC/MME/AMF hosting the cell, latitude, longitude and location parameters instead of the manual loading of such information since the latter might be error prone. Especially for the 5G case, static information about the equipment served in specific areas (either UEs or IoT sensors) can be added as well.

After having installed the servers, the database and the SON platform software modules must proceed with the final IP planning of the system, ensure connectivity to the operators' data network, and enable connectivity with the OSS assigned to handle each technology, as depicted in Figure 6. This figure is an extension of Figure 4 and includes the coexistence of a UMTS and an LTE network. In case a 5G network is included, the design rationale is identical to an AMF interacting with the gNBs, the 5G OSS and the 5G KPIs database (db). Finally, the initial tuning of the platform and applications according to the desired policy must be performed before starting the system.

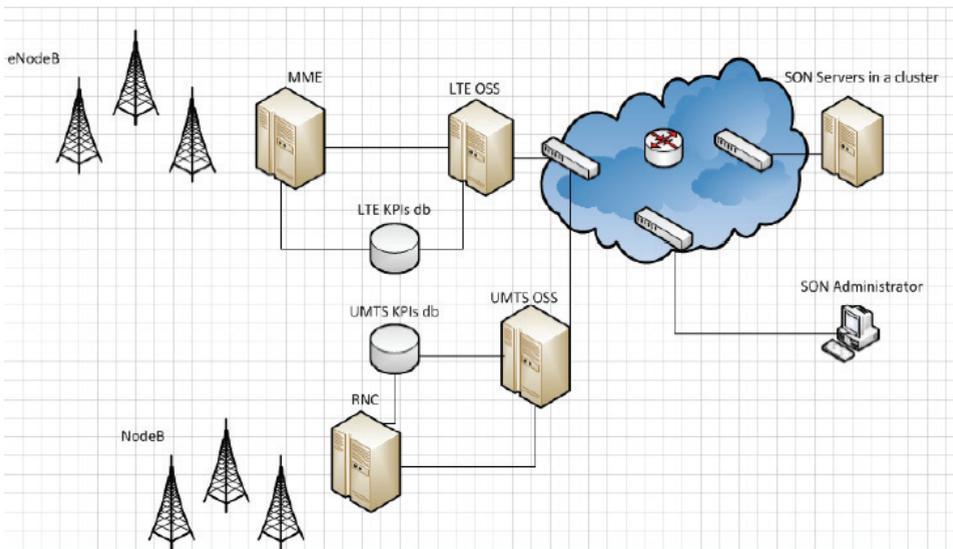


Figure 6. Interconnection of SON server cluster with UMTS and LTE OSS (centralized architecture).

### 3. SON Applications

#### 3.1. Basic SON Applications in 2G/3G/4G/5G Networks and Operation Rationale

As far as the key applications for existing mobile networks are concerned, 3GPP and NGMN overlap in some of them; however, the major ones are the following [21–23,27,40,41,57–64]:

1. Automatic neighbor relation (ANR).
2. Automatic radio network configuration (initial radio and core parameters setting).
3. Mobility Load Balancing optimization (MLB).
4. Coverage and capacity optimization (CCO).
5. Mobility Robustness Optimization (MRO).
6. Automated configuration of physical cell identity (Cell ID (CI) and PCI optimization).
7. Interference reduction.
8. Minimization of drive testing (MDT).
9. Energy saving (ES).

10. RACH (Random Access Channel) optimization.
11. Inter-cell interference coordination (ICIC).

We believe that the most popular applications are ANR, MLB and CCO; thus, we explain their operation rationale, the factors that led to the need for them, and the issues that they resolve.

Before the development of SON systems, configuration, optimization, and healing processes were performed manually based on a time-consuming and error-prone lifecycle. As an example, problems such as interference, capacity, and coverage holes mitigation were identified through KPIs monitoring, CM/PM data collected from the OSS, alarms collection, drive test campaigns or even subscriber complaints. The next step was the manual analysis of KPIs and drive test results, and as a last step the network parameters reconfiguration and the monitoring of the new KPIs after the changes were committed to the network. In some cases, drive testing repetition was needed as well for a new evaluation.

As expected, these procedures frequently led to time-consuming and error-prone actions, especially in the cases of optimization and healing since capacity, coverage, and interference must be faced in real time and with a holistic approach. Moreover, the results after manual network parameter changes cannot be regarded as beneficial in all scenarios since they trigger a chain reaction of additional needed network changes for the neighboring cells, especially in dense areas. As expected, OPEX spending occurs as well.

As explained earlier, all SON applications are ruled by the Snapshot–Action–Feedback lifecycle. At this point, it should be noted that we can configure and tune the trigger thresholds of the snapshot phase so that the frequency of SON activities in the OSS is increased or decreased. Very frequent activities lead to system and OSS load, while rare activities do not have a fast outcome; thus, trigger balance is a key point.

### 3.2. The Key SON Applications Leading to Full Automation

#### 3.2.1. SON (ANR/Handover Success Rate) and NCL (The Need behind ANR)

Prior to SON applications, academy and industry proposed solutions for automating key optimization activities for network performance KPI advancement, mainly related to the handover success rate and load balancing. The handover success rate enables continuous connection for User Equipment (UE) while subscribers are moving and it is one of the most critical KPIs for all mobile networks radio access technologies. A neighbor cell list (NCL) includes handover candidate cells and all UEs connected to a specific cell are informed about this candidate list [66]. More specifically, NCL includes interactions among:

- Intra-frequency (same technology and same frequency) cells of the serving and neighbor cells.
- Inter-frequency (same technology and different frequency) cells of the serving and neighbor cells.
- Inter-RAT (different radio access technology) cells of the serving and neighbor cells.
- All the above cell interactions of the serving and neighbor cells in the same Base Transceiver Station (BTS)/NodeB/eNodeB/gNB.

Before SON, neighbor cell lists (NCL) were manually configured by radio network planning and optimization engineers, by using data from the current and real-time status of the network (OSS) combined with data from coverage simulation tools to extract the optimal list. Apart from being time consuming and the need for operational expenses, this manual process includes multiple drawbacks, since it is very difficult to predict the radio propagation environment (at the microcell level) and predict all the actions that might take place such as the operation of new cells not activated during the manual NCL creation process or the non-operational status of specific cells at that time [67,68]. In [69], the authors propose an automatic optimization algorithm based on base station coordinates and cell direction, but the disadvantage was that large lists of candidate cells were created, while the authors of [70,71] were focused on finding potential neighbors based on cell coverage overlap.

The prementioned proposals had a key contribution to the road towards self-optimization and their common disadvantage was that static network information data were mainly used as input (coordinates and cell coverage according to simulation tools). This creates difficulties in adapting in a constantly changing and dynamic network environment; thus, the authors of [68] further included live measurements for the creation of an NCL.

The authors of [66] proposed a method for the self-configuration and self-optimization of NCLs. In the former case, the serving cells of the base stations collect signal quality information about the neighboring cells, while in the latter case real-time reports regarding neighboring cell quality measurements are collected from UEs and reported back to the serving cell. One of the key findings for the self-configuration scenario was that the neighbor’s pilot signal quality cannot be measured in the whole serving cell coverage area since a sectorized antenna is used; thus, live measurements from the moving UEs are needed. A simulation deployed predicted an 85% call success rate for self-configuration and 97.3% for self-optimization. We believe that this result is optimistic; however, said paper is an excellent contribution to the explanation of self-optimizing and self-organizing rationale and a prelude to SON ANR applications.

Figure 7 depicts the rationale of the self-configuration and self-optimization as discussed prior to the official launch of SON ANR applications.

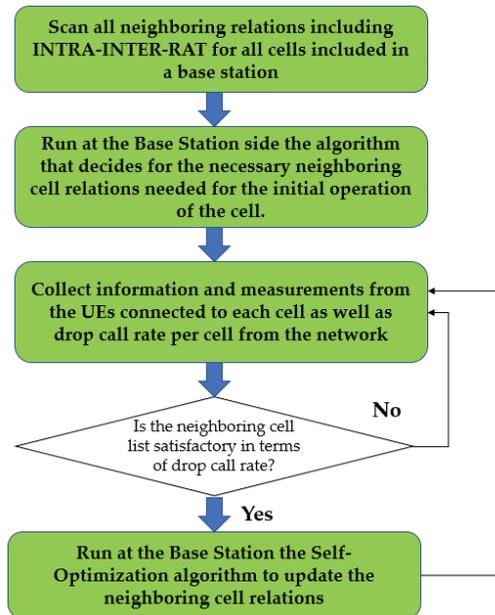


Figure 7. Self-configuration and self-optimization process.

### 3.2.2. SON—ANR

Possibly the most common self-optimization application is ANR (Automatic Neighbor Relations), which replaces the manual process of neighbor cell configuration and handover optimization previously described by creating automatically and in real time the most optimal neighbor relation list for a cell operating in either technology (2G/3G/4G/5G) and in relation to either technology as well. The application runs both for operational and new cells added into the network and it is a key use case for B5G development since recent publications analyze and propose hybrid solutions for ANR implementation in 5G networks [64,72].

SON ANR applications are mainly used to create intra-frequency (same technology and same frequency) and inter-frequency (same technology and different frequency) neigh-

bor lists since these are of high importance; however, Inter-RAT (Radio Access Technology, different technology and different frequency) relations can be optimized too. ANR applications act by deleting existing redundant neighbor relations that either are not used or exist with cells out of the coverage area and they add important neighbor relations according to the status of the network at a given time. This means that even if a neighbor relation is deleted because it has low usability rates, the same one might be added again after a certain period since subscribers moving around the coverage area of a cell might need this relation again so that handovers in either technology are successful. All cells have a maximum allowed number of neighbor relations that can be set and ANR applications are initially tuned during the initial SON system tuning process to keep the specific rule according to the policy that each network operator demands. The ANR operation is repeated multiple times until the neighbor list is perfectly optimal according to real-time metrics and according to the subscriber's mobility and distribution.

### 3.2.3. Input for ANR and Triggering

The following thresholds and parameters are set as input to the SON ANR algorithm, which evaluates the existing neighbor list and creates a candidate list for deletion and addition actions [66–72]:

- DCR (Drop Call Rate) for a defined sample period or bad quality (SNR ratio).
- Minimum number of calls served (low or high usability of a cell).
- Importance and usability of existing neighbor relations according to the ratio of handovers for a specific neighbor relation compared to the total number of handovers of the cell.
- Missing neighbor events (collection of network counters reporting missing neighbors).
- Intersection (coverage pattern of each cell and overlap among them).
- Cell coverage including propagation delay counters collected from the OSS.
- Layer Management Strategy (LMS): Policy followed by the operator regarding the inter-frequency relations (i.e., three carriers are used (F1, F2, and F3) if a handover is allowed).

### 3.2.4. Mobility Load Balancing (MLB) and Traffic Steering (TS)

In a similar manner, Mobility Load Balancing (MLB) or traffic steering (TS) is one of the key optimization activities applied either independently, or in case multiple radio access technologies coexist. The target of MLB is to forward voice and data traffic to the most appropriate frequency/carrier or hierarchy cell layer or radio technology according to the radio parameters policy followed by the network operator. MLB can be applied either in idle or in connected mode states of a UE (inter-/intra-frequency or Inter-RAT scenarios).

In GSM/UMTS networks, static load balancing procedures usually take place while in LTE and 5G mechanisms for MLB are embedded in 3GPP standardized SON functions, leading to autonomous operation based on network counters' feedback [57–64,73]. Basic parameters that affect traffic steering/load balancing concern neighbor cell level parameters related to intra-/inter-frequency or inter-system relations or cell parameters that concern only a specific cell [22,23]. Currently, key mechanisms for traffic steering include the tuning of inter- or intra-frequency handover or reselection parameters, absolute priorities (AP) (the UE is informed in idle mode about the priority of a candidate camp frequency or candidate RAT) and Basic Biasing (BB) (triggering cell selection or reselection in idle mode).

As a definition, network load can be described by the following scenarios with the first one being the most common [8,22,23,64,73]:

- Radio resource load related to blocking due to a lack of resources.
- BTS/NodeB/eNodeB/gNB hardware resource load.
- Transport/backhaul network resource load.

### 3.2.5. Scenarios and Impairments That Trigger MLB Operation

Depending on the optimization scenario that must be deployed, load balancing and traffic steering aim to resolve the following impairments [8,22,23,64,73]:

- Offload loaded cells towards non-loaded ones, achieve a better UE throughput and QoS (Quality of Service), and ensure optimum cell capacity while users are moving within an area of interest, as a scenario assumes that an operator uses three carriers for one radio access technology. When a high load is observed in one of the three carriers, load balancing is achieved by forwarding traffic from one of the three carriers to the other through a reselection or handover.
- Offload higher power/capacity macro cells towards low-power/capacity micro/pico or femto cells. This scenario is deployed through the decreasing or increasing cell coverage by adjusting cell power. The same stands in a case where cells operate at high power, but traffic is low.
- Ping pong effects limitation (cases where handovers take place constantly). This phenomenon is seen especially in cases such as highways with fast moving UEs trying to camp on lower cells.

### 3.2.6. SON MLB and TS Applications—Conflicts among MLB and MRO (Mobility Robustness Optimization)

As expected, the manual handling of load balancing and setting of optimum handover and reselection parameters in real time is extremely difficult due to the constant mobility of users that cannot be predicted. The only exception concerns the cases of mass event handling such as planned concerts or athletic events where operators can approximately predict the load/capacity demands of the network and plan/tune the network accordingly by using specific network features or by adding extra equipment (mobile base stations). Based on these factors, SON MLB applications are considered as one of the most critical SON functions, while their impact has been regarded as one of the most basic for UMTS and LTE networks [22,23]. The same stands for 5G as well. MLB operation can overcome load balancing issues but needs careful tuning since rapid MLB parameter changes increase the Physical Uplink Shared Channel (PUSCH) interference level in the uplink channel and decrease average the Channel Quality Indicator (CQI) in the downlink for the LTE case [74].

Furthermore, SON MLB goes hand in hand with another application called SON MRO (Mobility Robustness Optimization), an application that aims to optimize cell reselection in idle mode and handovers in connected mode. However, it has been noticed that MRO operation creates conflicts with MLB and vice versa, since the same mobility parameters are handled by both SON functions. As an example, after MLB changes, handover performance degradation might be noticed and, as expected, the MRO application finds that user mobility problems exist [22,23,75]. Figure 8 depicts the conflicts that might be created in mobility thresholds (idle and connected modes) through MLB and MRO applications since both applications interact with idle and connected mode thresholds, and thus conflicts can easily occur.

The authors of [76] proposed a solution so that these conflicts are avoided through a scheme where MLB is based only on cell reselection instead of handover since there is no conflict between the cell reselection parameters changed by MRO and MLB. Instead of changing mobility thresholds, an alternative approach for MLB applications which is widely followed by SON vendors is based on the changing of the antenna tilt (either down-tilt or up-tilt) since this action affects the coverage area of cells and traffic can be shifted to a neighbor cell that is less loaded. Additionally, the authors of [77] approached MLB operation through antenna tilt changes and concluded on the importance of antenna vertical beamwidth and the addition of an extra handover offset so that a minimum difference in the received power between the serving and neighbor cells is ensured before handover. No matter which technique is used, the aim is to enable subscribers to move and camp in non-congested cells.

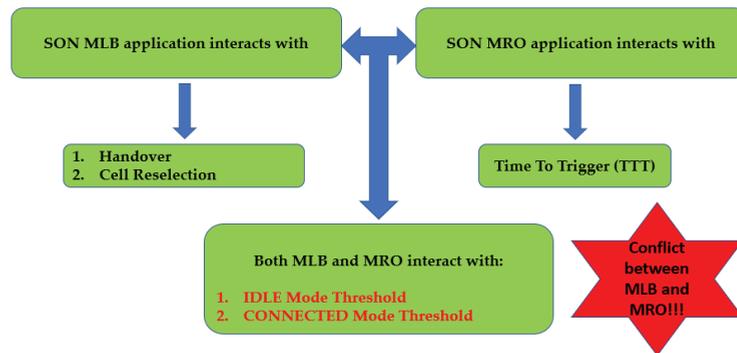


Figure 8. Conflicts between MLB and MRO.

Future research for 5G includes the determination of load by considering the backhaul state as well. A variety of MLB and MRO algorithms have been proposed; however, the former might fail in case backhaul capacity limitations occur. This is very critical for future 5G networks where base station backhaul limitations might lead to low rates for bandwidth intensive services [78]. Finally, the authors of [79] presented a SON load balancing algorithm that considers the backhaul restrictions and proposes the future consideration of backhaul restriction in SON MRO algorithms.

### 3.2.7. SON Coverage and Capacity Optimization (CCO), Interference Management and Adaptive Antennas

SON Coverage and Capacity Optimization (CCO) and Interference Management applications were developed to ensure uniform coverage and capacity in problematic areas where continuous coverage cannot be ensured, signal quality is not satisfactory, and interference is a key issue that cannot be handled easily manually in real time. The aim of this is to extend a desired cell’s coverage, mitigate coverage holes, and avoid interference at the same time. Triggered mainly by the high drop call rate, low handover success rates, and deteriorated QoS KPIs such as SINR (Signal to Interference Ratio), the specific applications operate mainly through antenna electrical tilt adjustments (adjusting azimuth and main or side lobes) centrally through the network OSS by controlling the Remote Electrical Tilt (RET) installed in each antenna [22,23]. Antenna tilt changes (either down-tilting or up-tilting) must be performed very carefully since they affect coverage boundaries, while an interference issue might be triggered after such a change.

- When electrical tilt is decreased (uptilt), the coverage area of the cell might overlap with a neighboring cell, and thus interference is created together with capacity issues. This specific case creates the so-called “over-shooting cells”.
- On the other hand, when electrical tilt is increased (down-tilt), the coverage area of the cell is reduced, leading to interference mitigation towards the neighbor cells, traffic offloading because of capacity limitations, and the elimination of over-shooting cells. This tradeoff must be constantly monitored, and operators must perform such changes in a central manner through SON CCO application.

Antenna parameter optimization needs constant monitoring; thus, it can be said that electrical tilt changes through SON and the algorithms supporting the specific applications need to react very fast in frequent load cases. Alternative methods include changes related to cell output power or reference/pilot signal power.

Apart from central control through RET, another option is the usage of Active Antenna Systems (AAS), which are able to perform antenna beam forming or MIMO with spatial multiplexing, while the key behind the operational rationale of AAS is to create spatial beamforming patterns [80]. This approach is expected to be a key technique for 5G networks due to the importance of massive MIMO for 5G development. Finally, the authors of [81]

further analyzed the role of ML algorithms in antenna tilt changes and more specifically the role of reinforcement learning (RL) algorithms for obtaining the optimum result. The role of specific algorithms not only in the CCO case but SON as a system seems to present an ideal application scenario according to the upcoming conferences during 2022 on the issue, but in our opinion the use of deep reinforcement learning seems even more challenging and can bring even better results [82].

#### 4. Machine Learning Algorithms for SON

The application of ML and AI techniques in future wireless networks constitutes a separate and very wide research field, but to address this we must refer briefly to the key algorithms applied in SON systems. In this section, our target is to briefly describe the main machine learning (ML) algorithm taxonomy applied in SON systems. Machine learning (ML) as a subset of artificial intelligence (AI) is a key enabler for SON systems since all related applications are based on intelligent ML algorithms that use real network data (counters and KPIs) as input to make decisions about the action needed for each optimization issue and create executable scripts for the OSS.

Future research on SON systems for 5G networks includes the development of new ML algorithms that can adapt in a variety of scenarios and provide cognitive behavior and intelligence in past and present network statuses, so that full end-to-end automation through SON can be achieved [30]. Table 2 describes the suggested ML technique and possible algorithm that can be used per SON application.

**Table 2.** ML technique and possible algorithm used per SON application.

SON Application	Machine Learning Technique	Algorithms That Can Be Used
Mobility Load Balancing (MLB)	RL or DRL, UL, SL	Q-learning
		Fuzzy Q-learning
		Dynamic Programming
		K-means clustering
		Polynomial regression
Mobility Robustness Optimization (MRO)	RL or DRL, UL, SL	Q-learning
		Fuzzy control
		Pattern identification SOM
		Semi-Markov model
		ANN
Coverage and Capacity Optimization (CCO)	RL or DRL, UL	Fuzzy Q-learning
		ANN

##### 4.1. ML Algorithms Taxonomy

ML taxonomy in computer science is usually split into supervised learning (SL), unsupervised learning (UL) and reinforcement learning (RL). According to the approach of the authors in [15], SON systems’ learning algorithms can be classified into supervised and unsupervised learning, similarly to humans, who learn something independently or with the help of a teacher. Figure 9 depicts the taxonomy based on SL, UL, and RL.

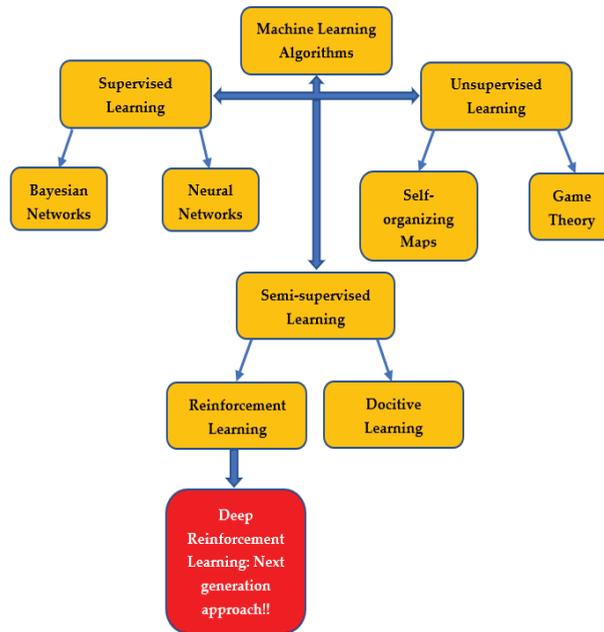


Figure 9. ML Algorithms taxonomy.

#### 4.1.1. Supervised Learning (SL)

Supervised learning is an ML technique based on training data used as input and test data leading to an output prediction close to reality [9]. A huge number of algorithms proposed can be found in the literature; however, the most common ones related to self-optimizing network functions are the following:

- Bayesian: Bayesian algorithms are based on Bayes theorem and their aim is to calculate probabilities according to pre-existing probabilities. Additionally, they do not require a large training sequence.
- Neural networks: This specific category comes from biology (brain neural networks). Neural networks are trained by taking examples and generating identification characteristics from the training input that they are given. A neural network consists of a set of interconnected nodes, while the processing ability of the nodes' network is described through the node connection weights, created by the adaptation of a learning process obtained by training patterns provided as input [83].

Additional SL algorithms categories include:

- Decision trees (DT).
- Hidden Markov Models (HMM).
- Support Vector Machines (SVMs).
- K-nearest neighbors (k-NN).

#### 4.1.2. Unsupervised Learning (UL)

Unsupervised learning does not involve any training input or sample contrary to supervised learning. The key rationale of unsupervised learning is to recognize a pattern among the input data that the algorithms receive so that future inputs can be predicted. Typical examples include social networks analysis while in SON applications. Unsupervised learning algorithms are used mainly for self-healing and self-optimizing use cases [9]. The most common cases are the following:

- Self-organizing maps (SoM): A SoM intends to identify groups of data so that a representation of the input data is created. This technique is called clustering. The main idea behind SoM is to group cells with similar network parameter settings and trigger changes for the specific group uniformly.
- Game theory: Known since the 1950s and applied later in biology, social science and economy/political sciences during 1970s game theory have been used for network systems modelling. According to game theory, several players, nodes or artificial agents (in the case of computer science) perform actions that affect each other while the target is to get a zero-sum result by subtracting the losses of one player and adding the gains of the winning player [83]. According to the authors of [15], operators can avoid zero human interaction by using game theory approaches in mobile networks but limited literature concerning game theory applicability in LTE networks exists.

#### 4.1.3. Reinforcement Learning (RL)

This specific subcategory has been already proven as effective for autonomous vehicles, network elements routing, and other applications, including self-optimization. The basic rationale is based on learning through interactions with neighbor nodes. In more detail, the specific algorithms receive as input a reward function indicating that they work as expected [9,15]. Reinforcement learning is tightly coupled with Markov Decision Processes (MDP) in terms of defining a possible set of network states and a possible set of actions for each state. Possibly the most common examples are Q-learning and Fuzzy Q-learning. The specific algorithms are based on the rationale that for any finite Markov decision process (MDP), an optimal policy for awarding or maximizing the expected value of each step is identified. Deep reinforcement learning is the evolution of RL that seems to be an even more promising technique for 5G and beyond networks [84–88].

Reinforcement learning (RL) and deep reinforcement learning (DRL) as an advancement of RL is a very promising approach in the field of machine learning for B5G networks dealing with sequential decision making. The difference between the two relates to the fact that the former is based on dynamic learning with a trial-and-error method, while the latter is learning from existing knowledge and applies it to a new data set. However, the operation rationale is very similar and they are expected to have a key role in future telecommunication systems.

RL is vital due to the ability that can be provided to the agents to learn in real time with no previous knowledge of the environment and continuously interact with it, contrary to the other legacy ML techniques that require full knowledge of the environment and a training set as input, provided by an external resource to the algorithm. The main rationale is that an agent, which in our case might be a UE or a base station (BTS/NodeB/eNodeB/gNB), learns the optimum behavior by collecting information in real time and tries to obtain the maximum reward at each algorithm step [12,84–88]. The key terms related to RL algorithms are the following:

- Environment (e): A real-life scenario that an agent must interact with.
- Agent: An entity or a node that performs specific actions in an environment in order to gain some reward.
- State (s): The current situation in the environment.
- Reward (R): A return provided to the agent when a specific action is taken.
- Policy ( $\pi$ ): The rationale that the agent follows in order to decide on the next action according to the current state.
- Value (V): The long-term reward to the agent compared to the short-term reward.
- Value Function: The value of a state as a total amount of reward gained by the agent.
- Q value (Q) or Action Value: An estimation of how good it is to take the action at each state.

Figure 10 depicts the interaction between the environment and the agent and correlates the action, state, and reward terms related with RL.

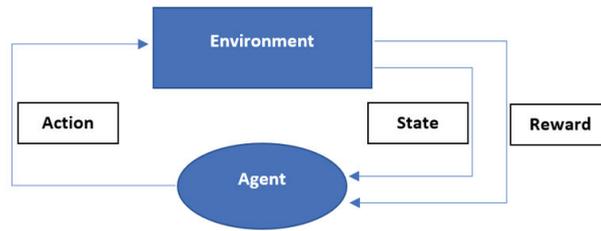


Figure 10. Environment and agent interaction.

Apart from the fact that the agent interacts with an uncertain environment and must face it in a holistic manner, one of the key challenges related to reinforcement learning is the exploitation and exploration tradeoff. This specific tradeoff is related to the selection of the actions already selected in the past from the agent proven to provide a beneficial reward and at the same time explore the environment so that even better actions are selected.

We cannot avoid referring to the ancestor of RL, the Bellman equation, which is a functional equation that describes the behavior of a dynamically changing environment over time and led to the development of dynamic programming methods that try to handle optimal control scenarios in such environments [88]. The Bellman equation describing the values received by an agent during the process use for trying to find the most optimal action to take at each step can be written as:

$$V(s) = \max [R(s,a) + \gamma V(s')] \tag{1}$$

The key terms of the equation are:

- Action performed by the agent is “a”.
- State instance by performing an action by the agent is “s”.
- The reward received for each beneficial action or not is “R”.
- The discount factor is Gamma “γ”.
- V(s): Corresponds to the value calculated at a specific point.
- R(s,a): The reward at a particular state after an action is performed.
- γ = Corresponds to the discount factor that determines if the agent as regards the beneficial the rewards received in the near future compared to those in the immediate future.
- V(s') = The value obtained at the previous state.

As far as 5G networks are concerned, recent approaches and proposals involving RL algorithms regard power and interference control and they can be a perfect match for the development of intelligent B5G SON platforms [12,84–87]. The UEs and RAN nodes can run RL and DRL algorithms so that real-time network optimization activities can be fully automated and able to be effective in real-time and continuously changing radio propagation environments.

#### 4.2. Docitive Learning (DL)

Docitive learning algorithms (from the word docere = to teach or transfer information) are mature ML algorithms based on the rationale that a decentralized network of nodes can exhibit a better performance in case nodes exchange information among themselves. Studies have shown that docitive learning can be effectively applied in cognitive radio (CR) networks as well [89].

### 5. Future SON Research Directions for B5G

Currently, SON research and development is spread over many different domains. Academy, industry, and worldwide standardization bodies such as 3GPP/ITU have evolved so that SON is fully embedded in future B5G solutions [8–14,20–23,30,56,64,72,73].

Interesting cases include:

- The development of Virtual SON (V-SON) platforms since NFV and SDN architectures have been suggested as one of the key drivers for SON deployment in 5G networks as well as the binding of SON with C-RAN.
- The development of new ML algorithms and the binding of SON with big data and deep learning technologies.
- The binding of SON with key 5G eMBB enablers such as mmWave.
- The application of SON in IoT network infrastructures, and massive Machine-Type Communication (mMTC).
- The backhaul network management through SON.
- The risk assessment and the development of security solutions and policies related to SON systems in 5G networks.
- The development of hybrid SON solution targeting in URLLC services.

Figure 11 depicts the key entities of a non-roaming 5G architecture as defined by 3GPP in [73]. The basic ones depicted in the figure are the following:

- Network Slice Selection Function (NSSF).
- Network Exposure Function (NEF).
- Network Repository Function (NRF).
- Policy Control Function (PCF).
- Unified Data Management (UDM).
- Application Function (AF).
- Network Slice Specific Authentication and Authorization Function (NSSAAF).
- Authentication Server Function (AUSF).
- Access and Mobility Management Function (AMF).
- Session Management Function (SMF).
- Service Communication Proxy (SCP).
- User Equipment (UE).
- (Radio) Access Network ((R)AN).
- User Plane Function (UPF).
- Data Network (DN), e.g., operator services, Internet access or third party services.

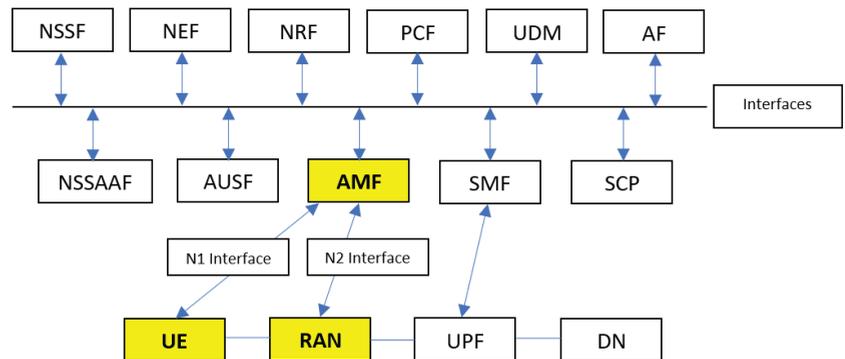


Figure 11. Non-roaming 5G architecture.

Among the key elements depicted in Figure 11, the ones directly interacting with 5G SON platforms shall be AMF, UE (Might Include UEs or IoT sensors), and RAN, including the gNBs.

### 5.1. NFV and SON (vSON) for 5G Networks—RAN Virtualization/C-RAN Architectures and SON

The telecommunications industry managed to develop critical network function entities such as Evolved Packet Core (EPC) in the case of LTE which is an indication that it can

be applied in 5G network entities as well [90,91]. An example is the migration from virtual Evolved Packet Core (vEPC) the Stand-alone for 5G core (SA) [92]. Since very critical parts of a network can be virtualized and legacy hardware tends to be replaced by virtualized solutions, SON platform virtualization over NFV seems to be intelligent, cost effective and fast in terms of deployment; however, performance metrics must be carefully examined since the interaction of a SON system with the OSS of a mobile network in real time and in terms of time response is very critical. Companies such as Cellwise have already deployed virtualized SON solutions [93].

Virtual SON (V-SON) as an extension of the NFV and SDN architecture has been suggested as one of the key drivers for SON deployment in B5G networks. This can occur either in a centralized or in a distributed/hybrid architecture approach, enabling the provision of SONaaS (SON as a service) [8,11,94,95].

The basic architectural functional blocks of an NFV architecture are the following [96]:

- Virtualized Network Function (VNF): VNF refers to a standard network function, non-virtualized prior to NFV introduction, which can be fully or partially virtualized. Typical examples might be the MME or AMF, the SGW (Serving Gateway), or even the eNodeB/gNB. In our case, SON might be one of the VNFs [95].
- Element Management (EM): Refers to the management operations strategy of the VNFs.
- NFV Infrastructure (NFVI): VNFI refers to the common hardware resources (servers and storage) that host the VNFs. Routers, switches and wireless links interconnecting the main servers can be regarded as part of the NFV infrastructure as well.

Moreover, C-RAN (Cloud-RAN), or Centralized-RAN, was first proposed as an architecture by China Mobile Research Institute in April 2010 providing a unified, centralized, and cloud computing solution, leading to the virtualization of the RAN part of 2G, 3G, 4G, and 5G networks [97].

Network operators can benefit from C-RAN since it can provide higher spectrum efficiency, interference, and capacity management, load balancing/traffic peaks handling, energy consumption, and CAPEX/OPEX savings [97–100]. The former is the outcome of the centralized or partially centralized operation of a Base Band (BB) pool gathering basic RAN functions and serving multiple radio units. The possible architectures are initially based on the assignment and sharing of the functionalities of the BBU (Baseband Unit) and Remote Radio Head (RRH) (referred as Remote Radio Unit (RRU) as well).

There are two basic approaches: In the first approach full centralization is provided, since the baseband unit provides layer 1, layer 2, and layer 3 functionalities, while in the second approach (partial centralization) the RRH provides the BBU functionalities leading to partial centralization [97–100]. The traditional strong relation of the BBU with the RRU where the latter coexist in the same site does not exist anymore since an RRU might belong to different BBUs and the functionality of a “Virtual BTS” is introduced. This is the key correlation point of C-RAN with Network Function Virtualization (NFV).

In our opinion, the combination of C-RAN with C-SON is a key enabler for future B5G networks, while with the rise of C-RAN it is possible that additional scenarios and use cases for C-SON systems might appear. The rationale behind this is that C-SON systems shall be able to monitor in a more centralized manner metrics and parameters related to the Baseband Units hardware operation and possibly new SON applications focusing on hardware management can be developed. An example might be the efficient hardware resource allocation assignment through C-SON applications running on the BBU pool which is a well-known issue for operators [91,101].

Finally, since network slicing allows the creation of multiple logical networks in the NFV domain able to simultaneously run on top of a shared physical network infrastructure, end-to-end virtual networks that include both networking and storage functions can be created. Network resources are partitioned and scenarios with 5G verticals with different latency and throughput demands can be created. In our opinion, this discrimination can be combined with SON functions per scenario.

### 5.2. Empowering SON for 5G with Big Data and New ML Algorithms

As we move towards 5G, the amount of training data needed for algorithms increases exponentially; thus, we conclude in dig data structures that must be handled. New ML algorithms must be developed so that SON systems are able to have an end-to-end vision of the network and conflicts among different SON applications handling the same or similar network parameters are resolved. The performance of ML algorithms depends on the representations that it learns to output. This is the point where recent approaches in deep learning such as “Representation learning” provide an efficient solution by transforming input data and defining if input training data such as specific network parameters represent a specific set mapped directly to the output. The core idea of representation learning is that the same representation may be useful for multiple tasks as an output while training sequences are classified in a more efficient manner as far as the output is concerned in this way [102].

The authors of [10] note that existing SON systems lack end-to-end network knowledge, and thus they proposed a big data framework (BSON) as a holistic approach including, apart from the already used network KPIs and performance counters, user/subscriber level data and user application-based data (social media and smartphone sensors).

The authors of [103] proposed an application-characteristics-driven SON system (APP-SON) already applied in a tier-1 operator for next-generation 5G networks. By combining a Hungarian Algorithm Assisted Clustering (HAAC) approach and a deep learning assisted regression algorithm, cell application characteristics were identified and used for SON optimization, leading to a better classification of useful KPIs per cell and a better QoE [103]. The experimental results prove that cell traffic can be profiled according to the applications used per user case. As a next step, weather conditions affecting the propagation environment shall be included.

In our opinion, the prementioned proposals could further evolve by combining information coming from geo-location systems already deployed in network operators [104].

### 5.3. SON, mmWave and Massive MIMO Technologies for 5G Networks

Compared with 3G and 2G technologies, 4G-LTE was revolutionary, but currently LTE networks are running out of bandwidth; thus, the spectrum usage of the millimeter-wave frequency range is a solution for 5G bandwidth requirements. mmWave is a part of the 5G standard and combined with 3D beamforming techniques, where multiple-element base station antennas use multiple antenna elements to form directional beams for transmission, and it is the key solution for providing low latency and high data speeds [8]. Furthermore, the propagation nature of mmWave leads to shorter communication distances, and thus base stations must be positioned closer if compared with existing mobile networks, leading to even more dense network clusters.

The binding of SON with mmWave and beamforming techniques is a recent very interesting concept and the authors of [13] proposed a use case related to directional cell search based on the rationale that the simultaneous transmission of broadcast signals reduces the amount of the needed cell search resources including latency during the process of recognizing neighboring cells. The tradeoff for this includes Radio Link Failures (RLFs) due to SINR deterioration (related to handover measurements) and smaller coverage. However, a centralized SON coordinator might improve the former.

Future research related to SON usage for directional cell search is also related energy savings in case beamforming is applied only in required areas so that coverage overlapping is avoided and with concepts such as the self-optimization of beam direction and transmission power.

Apart from mmWave, massive MIMO is also one of the key components of 5G networks and a key enabler for the provision of high-speed data rates in indoor and outdoor propagation environments operating either in mmWave frequencies or below 6 GHz [8]. ML algorithms can empower massive MIMO, leading to “Intelligent massive MIMO” solutions, and typical examples include massive MIMO power control and user positioning [105].

This means that ML based SON systems can be combined with massive MIMO and might be an interesting research scenario since beamforming can be controlled by SON platforms.

#### 5.4. SON for Backhaul Management in 5G

As already mentioned, the authors of [79] identified the backhaul restrictions and proposed a SON load balancing MRO algorithm considering the backhaul state. Since the application of SON in RAN proved to be very beneficial, academy and industry started working on the application of it in wireless backhaul so that self-configuring, self-optimizing, and self-healing capabilities are adopted in this domain as well. Even if radio access and backhaul technologies impose different handlings and have different characteristics, the operational challenges are closely related [106,107].

As an example, a self-optimization use case might enable cooperation with neighboring backhaul radios and interference mitigation, while the self-healing use case can adjust a link's transmission parameters in order to overcome a possible failed link or a new link addition [107,108]. In terms of self-healing, the key point is to automatically identify proactive rerouting needs in cases of performance degradation due to bottlenecks in the backhaul network. Inputs to backhaul SON might include QoS KPIs collected both from radio and backhaul networks [106].

Finally, the authors of [109] demonstrated a SON solution for mobile backhaul networks, leading to a 31% increase in traffic in case microwave link capacity degradation is detected and a 12% increase in case a link failure is detected.

#### 5.5. SON for IoT

SON applied in IoT infrastructures is a necessity as well and a key enabler for mMTC (massive Machine-Type Communications). IoT networks must be treated as large distributed systems and in this case, apart from neighbor sensor discovery, path establishment and service recovery, the key point is to apply SON functions for sensor energy management. The authors of [110] introduced a new IoT paradigm called Fog of Things (FoT), including FoT-Devices, FoT-Gateways and FoT-Servers, where IoT services have different profiles (FoT-Profiles) and they are offered in a distributed manner. Moreover, they proposed a platform called SOFTIoT enabling SON capabilities for FoT-Profiles, devices, and gateways.

Another very interesting work is described in [111], where the authors introduced a resilient, self-organizing middleware for IoT applications called SORRIR. The rationale of the system is to decouple resilience from business logic, monitor the applications operation, react to failures, and trigger automatically needed reconfigurations.

The authors of [112] investigated self-organization for low-power IoT networks through a lightweight distributed learning approach instead of legacy centralized optimized approaches implemented in the IoT devices. The aim was to reduce signaling between IoT nodes, which leads to decreased energy consumption and minimizes possible collisions.

Finally, in [113], industrial IoT load balancing through the scope of self-organization was investigated through a proposed load balancing scheme that considers wireless link quality, the congestion of the wireless channel, and the amount of data that need to be sent or received. The outcome is that through the specific scheme, reliability can be significantly improved since the packet drop rate is reduced by up to 85%.

#### 5.6. SON Platforms Risk Assessment and Security Concerns

Fifth-generation networks are exposed to the threats identified in 4G networks; however, the attack mitigation measures cannot be performed manually in a central manner due to the heterogeneity of the serving environment with M2M, D2D, and multi-equipment environments. Moreover, virtualization deployments raise trust issues between the operator and a Cloud service provider [114].

As far as SON platforms are concerned, some additional issues must be considered. The fact that a SON platform has full read/write execution rights to the OSS of each RAN

technology needs very careful consideration and monitoring, since the damage that might be created to the network infrastructure due to inappropriate and non-legitimate account usage can be irreversible.

The same stands for cases where a non-verified SON system manufacturer developers' code/script is imported into the system, since limited SON vendors enable through their APIs such actions and non-certified personnel, or small firms try to benefit from developing their own scripts due to the APIs provided with the system.

In our opinion, the latter might be acceptable in case of small and non-critical distributed SON architectures, but in case of centralized ones, a misconfiguration might be a complete disaster. Consider the case where wrong scripting is applied centrally in a SON system by non-certified developers by the manufacturer. The outcome shall be a vast number of OSS commands committed to the network which do not lead to the expected KPI advancement and are difficult to revert in real time, manually or by the SON system.

Finally, since SON systems are attached directly to the OSS, they must be completely isolated and reside behind all possible network security infrastructure such as anti-DDoS (Distributed Denial of Service), and NGFW (Next-Generation Firewalls) systems even if, typically, no web traffic (inbound or outbound) enters or leaves the SON system.

#### 5.7. SON for 5G Key Use Cases—The URLLC Use Case

Possibly the most difficult part to ensure and implement in 5G networks will be the URLLC use case. URLLC will be the key enabler for emerging applications and services including tactile internet, mobile factory automation, and inter-vehicular communications for improved safety such as autonomous driving. In order to achieve a successful deployment of the former applications, stringent requirements such as reliability, latency, and availability must be met [115].

Existing SON systems are mostly service agnostic; thus, the authors of [12] proposed SON usage as an approach for the successful implementation of the strict low-latency and reliability requirements through the selection of specific parameter sets as input to the system. The focus of this specific paper is on the Device to Device (D2D) technique, which regards mainly direct short-range communication MTC applications such as Wi-Fi Direct, Bluetooth Low Energy (BLE) or Near Field Communication (NFC).

On the other hand, Machine to Machine (M2M) communications regard mainly long-range Low-Power Wide-Area (LPWA) networks such as SigFox, Long-Range Wide-Area Networks (LoRaWAN) and Narrowband-IoT (NB-IoT), with applicability in smart cities, supply chain management, smart metering, security and surveillance, automotive communications, and e-healthcare scenarios, and it is expected that M2M traffic shall occupy around 45% of the total traffic of the Internet. Thus, currently operators focus on the deployment of M2M networks such as NB-IoT [116–119]. Based on these factors, we believe that future research on SON application in M2M applications is an interesting scenario that should be investigated.

#### 5.8. SON for 6G Networks

In the 6G era, energy efficiency, ultra-low latency, and high reliability shall be key issues to handle in standard mobile communication networks. Apart from this fact, in special cases, such as vehicular communications, the nodes must be able to perform self-organizing functions; thus, SON functions and network automation shall keep evolving [120,121]. As far as the next move towards 6G networks is concerned, and we are at the very early stages of research and standardization, SON systems must embed self-coordination capabilities to manage the complex relations between the different applications that are running, so that slow reactions in real-time situations are avoided. According to the authors of [122], hybrid SONs (H-SONs) combined with feedback from multiple feedback loops might be the ideal case. Virtualized, containerized, multi-tenant architectures as well as federated transfer learning and collective intelligence based on conventional AI techniques can improve the landscape as we move towards 6G architectures.

Another key technique as we move towards 6G regards O-RAN. O-RAN (Open RAN) is another key technique based on the rationale of splitting the centralized unit (CU), distributed unit (DU), and remote unit (RU). Intelligence in O-RAN architectures relies on the separation of the RAN Intelligent Controller (RIC) and allows one to gather radio resource management (RRM) and self-organizing network (SON) applications, controlling the radio resources and network. In addition to this fact, RIC is separated from the processing units and allows us to gather radio resource management (RRM) and self-organizing network (SON) functions, which control the radio resources and network. In the O-RAN concept, intelligence resides in the RIC, employing AI and ML models for radio network automation [123].

Finally, the prominent benefits of network slicing techniques are expected to be blended with SON functions as we move towards 6G infrastructures. As an example, a SON-based network management architecture spread across RAN and core network shall be necessary for controlling networks in a holistic manner with SON functions operating at the user and control plane and cooperating with others running at the core level [124].

## 6. Conclusions

Self-organizing network (SON) technologies constitute one of the key enablers for B5G network deployment and operation since network management, optimization, and maintenance processes are expected to become even more complicated because of the diversity of services and equipment and the escalation of the number of devices in the network. Without SON, these tasks will become unfeasible. Academia and industry managed to make network automation a reality through SON and embedded machine learning (ML) and artificial intelligence (AI) technologies in mobile networks, while the adoption of SON platforms by many operators worldwide over the last few years has yielded impressive results in terms of KPI advancement and tremendous changes in terms of network management processes.

In this paper, we presented and analyzed in detail the rationale of operation of SON systems and their evolution across generations, and proposed design methodologies for SON applications in B5G environments. We described in detail the design and the components of a C-SON system able to handle 4G and 5G networks and the dimensioning principles that must be considered prior to a SON system deployment in a RAN, including 2G/3G/4G/5G technologies. As far as the applications are concerned, we provided a deep insight into ANR, MLB, MRO, and CCO and explained their operation rationale.

Moreover, we described the key AI and ML algorithms needed for successful SON deployment and described the most recent related research directions such as RL and DRL. These specific algorithms are ideal for B5G SON systems due to their ability to interact with the environment and collect information about the next steps without the need for previous knowledge or training information. Finally, we highlighted the most important points that SON research should focus on in order to guarantee the successful application of SON solutions in future networks such as B5G or 6G. Key points regard the interaction of SON functions with NFV, Big Data, backhaul management, IoT infrastructures, important B5G technology enablers such as mmWave and massive MIMO, and finally the interaction with basic 5G use cases such as URLLC.

**Author Contributions:** Conceptualization, A.G.P. and G.C.P.; Methodology, A.G.P. and G.C.P.; Investigation and writing, A.G.P. and G.C.P.; Review and editing: A.G.P. and G.C.P.; Supervision, G.C.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable; the study does not report any specific data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Welcome to the Era of Connected Intelligent Machines. Available online: <https://www.ericsson.com/en/reports-and-papers> (accessed on 3 February 2022).
2. 5G Radio Access Technology and Capabilities. Available online: <https://gsacom.com/paper/5g-radio-access-technology-and-capabilities/> (accessed on 3 February 2022).
3. Rost, P.; Banchs, A.; Berberana, I.; Breitbach, M.; Doll, M.; Droste, H.; Mannweiler, C.; Puente, M.A.; Samdanis, K.; Sayadi, B. Mobile Network Architecture Evolution toward 5G. *IEEE Commun. Mag.* **2016**, *54*, 84–91. [CrossRef]
4. Hossain, E.; Hasan, M. 5G Cellular: Key Enabling Technologies and Research Challenges. *IEEE Instrum. Meas. Mag.* **2015**, *18*, 11–21. [CrossRef]
5. 5G Network Transformation. Available online: [https://www.5gamerica.org/wp-content/uploads/2019/07/5G\\_Network\\_Transformation\\_Final.pdf](https://www.5gamerica.org/wp-content/uploads/2019/07/5G_Network_Transformation_Final.pdf) (accessed on 3 February 2022).
6. SK Telecom’s View on 5G Vision Architecture Mobile World Congress, Barcelona. 2014. Available online: [https://assets.mwcbarcelona.com/SK-Telecom-5G-White-Paper\\_2019.pdf?mtime=20210505105924&focal=none](https://assets.mwcbarcelona.com/SK-Telecom-5G-White-Paper_2019.pdf?mtime=20210505105924&focal=none) (accessed on 3 February 2022).
7. LTE to 5G: The Global Impact of Wireless Innovation. Available online: <https://rysavresearch.files.wordpress.com/2018/08/2018-08-rysav-5g-americas-lte-to-5g.pdf> (accessed on 3 February 2022).
8. Rodriguez, J. *Fundamentals of 5G Mobile Networks*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2015.
9. Moysen, J.; Giupponi, L. From 4G to 5G: Self-Organized Network Management Meets Machine Learning. *Comput. Commun.* **2018**, *129*, 248–268. [CrossRef]
10. Imran, A.; Zoha, A. Challenges in 5G: How to Empower SON with Big Data for Enabling 5G. *IEEE Netw.* **2014**, *28*, 27–33. [CrossRef]
11. Pérez, M.G.; Pérez, G.M.; Giardina, P.G.; Bernini, G.; Neves, P.; Alcaraz-Calero, J.M.; Wang, Q.; Koutsopoulos, K. Self-Organizing Capabilities in 5G Networks: NFV & SDN Coordination in a Complex Use Case. Available online: <https://selfnet-5g.eu/publications/> (accessed on 3 February 2022).
12. Mandutz, E.R.; Wasway, A.; Thiely, A.M. Capacity Optimization for Ultra-Reliable Low-Latency Communication in 5G—The SON Perspective. In Proceedings of the IEEE 13th International Conference on Network and Service Management (CNSM), Tokyo, Japan, 26–30 November 2017.
13. Ahmed, F.; Deng, J.; Tirkkonen, O. Self-Organizing Networks for 5G: Directional Cell Search in mmWave Networks. In Proceedings of the 2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC) 2016; Institute of Electrical and Electronics Engineers (IEEE): Manhattan, NY, USA, 2016; pp. 1–5.
14. Sun, S.L.; Kaoch, M.; Ran, T.T. Adaptive SON and Cognitive Smart LPN for 5G Heterogeneous Networks. *Mob. Netw. Appl.* **2015**, *20*, 740–745. [CrossRef]
15. Aliu, O.G.; Imran, A.; Evans, B. A Survey of Self-Organization in Future Cellular Networks. *IEEE Commun. Surv. Tuts.* **2013**, *15*, 336–361. [CrossRef]
16. White Paper by HUAWEI and TELUS. Next Generation SON for 5G. HUAWEI Technologies. 2016. Available online: <https://www.huawei.com/en/technology-insights/industry-insights/outlook/mobile-broadband/insights-reports/next-generation-son-for-5g> (accessed on 3 February 2022).
17. Andrews, J.G.; Buzzi, S.; Choi, W.; Hanly, S.V.; Lozano, A.E.; Soong, A.C.K.; Zhang, J. What Will 5G Be? *IEEE J. Sel. Areas Commun.* **2014**, *32*, 32–1065. [CrossRef]
18. Road to 5G: Introduction and Migration, White Paper. April 2018. Available online: [https://www.gsmacom.com/futurenetworks/wp-content/uploads/2018/04/Road-to-5G-Introduction-and-Migration\\_FINAL.pdf](https://www.gsmacom.com/futurenetworks/wp-content/uploads/2018/04/Road-to-5G-Introduction-and-Migration_FINAL.pdf) (accessed on 3 February 2022).
19. CISCO Crosswork Optimization Engine. Available online: <https://www.cisco.com/c/en/us/products/collateral/cloud-systems-management/crosswork-network-automation/datasheet-c78-742113.html> (accessed on 3 February 2022).
20. NGMN Alliance. 5G Network and Service Management including Orchestration. V.3.14.0. Available online: [https://www.ngmn.org/wp-content/uploads/Publications/2019/190312\\_5G\\_Network\\_and\\_Service\\_Management\\_including\\_Orchestration\\_3.14.0.pdf](https://www.ngmn.org/wp-content/uploads/Publications/2019/190312_5G_Network_and_Service_Management_including_Orchestration_3.14.0.pdf) (accessed on 3 February 2022).
21. 3GPP TR 28.861, V0.2.0 (2018–11). Study on the Self-Organizing Networks (SON) for 5G networks (Release 16). Available online: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3556> (accessed on 3 February 2022).
22. Ramiro, J.; Hamied, K. *Self-Organizing Networks—Self-Planning, Self-Optimization and Self-Healing for GSM, UMTS and LTE*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2012.
23. Hamalainen, S.; Sanneck, H.; Sartori, C. *LTE Self-Organizing Networks (SON) Network Management Automation for Operational Efficiency*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2012.
24. Rango, F.D.; Veltri, F.; Tropea, M.; Santamaria, A.F.; Fazio, P.; Malfitano, A.; Marano, S. Interdisciplinary Issues for the Management of Next Generation Autonomic Wireless Networks Nature Inspired Techniques. *Int. J. Mob. Netw. Des. Innov.* **2007**, *2*, 141–152.
25. GSMA Intelligence. Global Mobile Trends 2021 Navigating COVID-19 and Beyond. December 2020. Available online: <https://data.gsmaintelligence.com/api-web/v2/research-file-download?file=141220-Global-Mobile-Trends.pdf&id=58621970> (accessed on 1 February 2022).

26. Mobile Networks: Automation Optimized Performance. White Paper by VIAVI. Available online: <https://www.viavisolutions.com/en-us/literature/mobile-networks-automation-optimized-performance-white-papers-books-en.pdf> (accessed on 3 February 2022).
27. NGMN. Use Cases Related to Self-Organizing Networks, Overall Description. 2007. Available online: [https://ytd2525.files.wordpress.com/2012/06/ngmn\\_use\\_cases\\_related\\_to\\_self\\_organising\\_network\\_overall\\_description.pdf](https://ytd2525.files.wordpress.com/2012/06/ngmn_use_cases_related_to_self_organising_network_overall_description.pdf) (accessed on 3 February 2022).
28. Laiho, J.; Wacker, A.; Novosad, T. *Radio Network Planning and Optimization for UMTS*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2006.
29. Ajay, R. *Mishra, Fundamentals of Network Planning and Optimisation 2G/3G/4G: Evolution to 5G*; John Wiley & Sons: Hoboken, NJ, USA, 2018.
30. Zappone, A.; Di Renzo, M.; Debbah, M. Wireless Networks Design in the Era of Deep Learning: Model-Based, AI-Based, or Both? *IEEE Trans. Commun.* **2019**, *67*, 7331–7376. [CrossRef]
31. Asghar, A.; Farooq, H.; Imran, A. Self-Healing in Emerging Cellular Networks: Review, Challenges, and Research Directions. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 1682–1709. [CrossRef]
32. SOCRATES. Available online: [www.fp7-socrates.eu](http://www.fp7-socrates.eu) (accessed on 3 February 2022).
33. SEMAFOUR. Available online: <http://www.fp7-semafour.eu/en/news/2015/08/31/semafour-final-report-available/> (accessed on 3 February 2022).
34. SESAME. Small Cells Coordination for Multi-Tenancy and Edge Services. Available online: <https://5g-ppp.eu/sesame/> (accessed on 3 February 2022).
35. SELFNET. Framework for Self-Organized Network Management in Virtualized and Software Defined Networks. Available online: <http://www.cognet.5g-ppp.eu/cognet-in-5gpp/> (accessed on 3 February 2022).
36. COGNET. Cognitive Networks. Available online: <https://5g-ppp.eu/> (accessed on 3 February 2022).
37. Gandalf. Monitoring and Self-Tuning of RRM Parameters in a Multisystem Network. Available online: <http://www.celtic-initiative.org/> (accessed on 3 February 2022).
38. Project OPERA-Net2. Available online: <https://www.celticnext.eu/project-opera-net2/> (accessed on 3 February 2022).
39. Befemto Project. Available online: [www.ict-befemto.eu](http://www.ict-befemto.eu) (accessed on 3 February 2022).
40. NGMN. Next Generation Mobile Networks Informative List of SON Use Cases. 2007. Available online: <https://fddocuments.in/document/ngmn-informative-list-of-son-use-cases.html> (accessed on 3 February 2022).
41. NGMN. NGMN Recommendations on SON and O&M Requirements. 2008. Available online: [https://www.ngmn.org/wp-content/uploads/NGMN\\_Recommendation\\_on\\_SON\\_and\\_O\\_M\\_Requirements.pdf](https://www.ngmn.org/wp-content/uploads/NGMN_Recommendation_on_SON_and_O_M_Requirements.pdf) (accessed on 3 February 2022).
42. Available online: <https://www.lightreading.com/mobile/self-organizing-network-son/verizon-completes-son-trials-with-cisco-ericsson-/d/d-id/718119> (accessed on 3 February 2022).
43. Available online: <https://www.cisco.com/c/en/us/about/corporate-strategy-office/acquisitions/hcltechnologies.html#~{}tab-overview> (accessed on 3 February 2022).
44. Cisco SON Mobile World Congress 2016 Presentation—CISCO SON Powered Business Solutions. Available online: <https://www.youtube.com/watch?v=6XhSveos13Y> (accessed on 3 February 2022).
45. SON Optimization Manager. Available online: <https://www.ericsson.com/en/portfolio/digital-services/automated-network-operations/network-management/son-optimization-manager> (accessed on 3 February 2022).
46. AI Enabling Intelligent Operations, White Paper by HUAWEI. Available online: <https://carrier.huawei.com/en/spotlight/ai-enabling-intelligent-operations> (accessed on 3 February 2022).
47. VIAVI 5G RAN Solutions. Available online: <https://www.viavisolutions.com/en-us/solutions/5g-ran> (accessed on 3 February 2022).
48. CELLWISE. Available online: <https://www.cellwize.com/resources/> (accessed on 3 February 2022).
49. TEOCO. Available online: <https://www.teoco.com/products-services/ran-solutions/network-optimization/> (accessed on 3 February 2022).
50. Real-Time Virtualized Network Automation & Optimization. Available online: <http://airhopcomm-web.com/products/eson> (accessed on 3 February 2022).
51. ELISAPOLYSTAR. Available online: <https://www.elisapolystar.com/automated-operations/> (accessed on 3 February 2022).
52. COMARCH. Available online: <https://www.comarch.com/telecommunications/oss-solutions/self-organizing-network/> (accessed on 3 February 2022).
53. 5G-PPP.eu. Available online: <https://5g-ppp.eu/wp-content/uploads/2021/05/AI-MLforNetworks-v1-0.pdf> (accessed on 3 February 2022).
54. PIWORKS. Available online: <https://www.piworks.net/Products#centralized-son> (accessed on 3 February 2022).
55. INNOVILLE. Available online: <https://www.innovile.com/solutions/self-organizing-network/> (accessed on 3 February 2022).
56. IWSON Workshop. Available online: <https://wnc2018.ieee-wnc.org/workshop/7th-international-workshop-self-organizing-networks-iwson> (accessed on 3 February 2022).
57. 3GPP TS 36.300 Version 8.9.0 Release 8, Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Overall Description. July 2009. Available online: [https://www.etsi.org/deliver/etsi\\_ts/136300\\_136399/136300/08.09.00\\_60/ts\\_136300v080900p.pdf](https://www.etsi.org/deliver/etsi_ts/136300_136399/136300/08.09.00_60/ts_136300v080900p.pdf) (accessed on 3 February 2022).

58. 3GPP TS 36.300 Version 9.4.0 Release 9, Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Overall Description 2010-07. Available online: [https://www.etsi.org/deliver/etsi\\_ts/136300\\_136399/136300/09.04.00\\_60/ts\\_136300v090400p.pdf](https://www.etsi.org/deliver/etsi_ts/136300_136399/136300/09.04.00_60/ts_136300v090400p.pdf) (accessed on 3 February 2022).
59. 3GPP TR 36.902 Version 9.2.0 Release 9, Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Self-Configuring and Self-Optimizing Network (SON) Use Cases and Solutions 2010-09. Available online: [https://www.etsi.org/deliver/etsi\\_tr/136900\\_136999/136902/09.02.00\\_60/tr\\_136902v090200p.pdf](https://www.etsi.org/deliver/etsi_tr/136900_136999/136902/09.02.00_60/tr_136902v090200p.pdf) (accessed on 3 February 2022).
60. 3GPP TR 36.902 Version 9.3.1 Release 9, Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Self-Configuring and Self-Optimizing Network (SON) Use Cases and Solutions 2011-05. Available online: [https://www.etsi.org/deliver/etsi\\_tr/136900\\_136999/136902/09.03.01\\_60/tr\\_136902v090301p.pdf](https://www.etsi.org/deliver/etsi_tr/136900_136999/136902/09.03.01_60/tr_136902v090301p.pdf) (accessed on 3 February 2022).
61. 3GPP TS 32.500, Version 11.1.0 Release 11. Self-Organizing Networks (SON) Concepts and Requirements. 2011-12 and Version 10.0.0, Release 10, 2010-06. Available online: [https://www.etsi.org/deliver/etsi\\_ts/132500\\_132599/132500/11.01.00\\_60/ts\\_132500v110100p.pdf](https://www.etsi.org/deliver/etsi_ts/132500_132599/132500/11.01.00_60/ts_132500v110100p.pdf) (accessed on 3 February 2022).
62. 3GPP TS 32.501, Version 12.1.0 Release 12. Self-Configuration of Network Elements. 2013-12. Available online: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=2032> (accessed on 3 February 2022).
63. 3GPP TS 32.500, Version 12.1.0 Release 12. Self-Organizing Networks (SON) Concepts and Requirements. 2014-12. Available online: [https://www.arib.or.jp/english/html/overview/doc/STD-T63V12\\_30/5\\_Appendix/Rel12/32/32500-c10.pdf](https://www.arib.or.jp/english/html/overview/doc/STD-T63V12_30/5_Appendix/Rel12/32/32500-c10.pdf) (accessed on 3 February 2022).
64. Self-Organizing Networks (SON) for 5G Networks (3GPP TS 28.313 Version 16.0.0 Release 16). Available online: [https://www.etsi.org/deliver/etsi\\_ts/128300\\_128399/128313/16.00.00\\_60/ts\\_128313v160000p.pdf](https://www.etsi.org/deliver/etsi_ts/128300_128399/128313/16.00.00_60/ts_128313v160000p.pdf) (accessed on 7 February 2022).
65. Feng, S.; Seidel, E.; NOMOR Research GmbH Munich, Germany. *Self-Organizing Networks (SON) in 3GPP Long Term Evolution*; White Paper; White Paper; Nomor Research GmbH: München, Germany, 2008.
66. Nguyen, V.M.; Clausen, H. Efficient Self-Optimization of Neighbor Cell Lists in Macrocellular Networks. In Proceedings of the 2010 IEEE 21st International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), Istanbul, Turkey, 26–30 September 2010; pp. 1923–1928.
67. Olofsson, H.; Magnusson, S.; Almgren, M. A concept for dynamic neighbor cell list planning in a cellular system. In Proceedings of the PIMRC '96—7th International Symposium on Personal, Indoor, and Mobile Communications, Taipei, Taiwan, 18 October 1996; pp. 138–142.
68. Mehdi, A.; Pål, K.; Gunnarsson, F.F.; Kallin, H.; Moe, J.; Zetterberg, K. Neighbor cell relation list and measured cell identity management in LTE. In Proceedings of the NOMS 2008—2008 IEEE Network Operations and Management Symposium, Salvador, Brazil, 7–11 April 2008; pp. 152–159.
69. Guerzoni, R.; Ore, I.; Valkealahti, K.; Soldani, D. Automatic Neighbor Cell List Optimization for UTRA FDD Networks: Theoretical Approach and Experimental Validation. In Proceedings of the IWS/WPMC, Aalborg, Denmark, 18–21 September 2005; Available online: [https://www.researchgate.net/profile/David-Soldani-2/publication/256298187\\_Automatic\\_neighbor\\_cell\\_list\\_optimization\\_for\\_UTRA\\_FDD\\_networks\\_Theoretical\\_approach\\_and\\_experimental\\_validation/links/60781a8f881fa114b40d2c1f/Automatic-neighbor-cell-list-optimization-for-UTRA-FDD-networks-Theoretical-approach-and-experimental-validation.pdf](https://www.researchgate.net/profile/David-Soldani-2/publication/256298187_Automatic_neighbor_cell_list_optimization_for_UTRA_FDD_networks_Theoretical_approach_and_experimental_validation/links/60781a8f881fa114b40d2c1f/Automatic-neighbor-cell-list-optimization-for-UTRA-FDD-networks-Theoretical-approach-and-experimental-validation.pdf) (accessed on 3 February 2022).
70. Soldani, D.; Ore, I. Self-optimizing Neighbor Cell List for UTRA FDD Networks Using Detected Set Reporting. In Proceedings of the 2007 IEEE 65th Vehicular Technology Conference—VTC2007, Dublin, Ireland, 22–25 April 2007; pp. 694–698.
71. Parodi, M.; Kylvaja, G.; Alford, J.L.; Pradas, J. An Automatic Procedure for Neighbor Cell List Definition in Cellular Networks. In Proceedings of the 2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Espoo, Finland, 18–21 June 2007; pp. 1–6.
72. Gorcin, A.; Cotanis, N. Hybrid automatic neighbor relations for 5G wireless networks. In Proceedings of the 2017 International Symposium on Wireless Communication Systems (ISWCS), Bologna, Italy, 28–31 August 2017; pp. 175–180.
73. System Architecture for the 5G System (5GS) (3GPP TS 23.501 Version 16.6.0 Release 16). Available online: [https://www.etsi.org/deliver/etsi\\_ts/123500\\_123599/123501/16.06.00\\_60/ts\\_123501v160600p.pdf](https://www.etsi.org/deliver/etsi_ts/123500_123599/123501/16.06.00_60/ts_123501v160600p.pdf) (accessed on 7 February 2022).
74. Ruiz-Avilés, J.M.; Toril, M.; Luna-Ramírez, S.; Buenestado, V.; Regueira, M.A. Analysis of Limitations of Mobility Load Balancing in a Live LTE System. *IEEE Wirel. Commun. Lett.* **2015**, *4*, 417–420. [CrossRef]
75. Zhiqiang, L.; Hong, P.; Xue, K.; Peng, M. Conflict Avoidance between Mobility Robustness Optimization and Mobility Load Balancing. In Proceedings of the 2010 IEEE Global Telecommunications Conference GLOBECOM, Miami, FL, USA, 6–10 December 2010; pp. 1–5.
76. Toshiaki, Y.; Komine, T.; Konishi, S. Mobility Load Balancing Scheme based on Cell Reselection. In Proceedings of the ICWMC 2012, Venice, Italy, 24–29 June 2012.
77. Bratu, V.I.; Beckman, C. Antenna Tilt Load Balancing in Self-Organizing Networks. *Int. J. Res. Wirel. Syst. (IJRWS)* **2013**, *2*, 17–22.
78. Osseiran, A.; Boccardi, F.; Braun, V.; Kusume, K.; Marsch, P.; Maternia, M.; Fallgren, M. Scenarios for 5G mobile and wireless communications: The vision of the METIS project. *IEEE Commun. Mag.* **2014**, *52*, 26–35. [CrossRef]
79. Tall, A.; Altman, Z.; Altman, E. Self-Optimizing Load Balancing with Backhaul-Constrained Radio Access Networks. *IEEE Wirel. Commun. Lett.* **2015**, *4*, 645–648. [CrossRef]
80. Passive and Active Antenna Systems for Base Stations of IMT Systems. Available online: <https://www.itu.int/pub/R-REP-M2334> (accessed on 3 February 2022).

81. Massaro, A.; Wellington, D.; Aghasaryan, A.; Seidl, R. Reinforcement Learning for Antennas' Electric Tilts Optimization in Self Organizing Networks. In Proceedings of the 2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Online, 13–16 September 2021; Institute of Electrical and Electronics Engineers (IEEE): Manhattan, NY, USA, 2021; pp. 1122–1127.
82. IEEE ICC, Tutorials. Available online: <https://icc2022.ieee-icc.org/program/tutorials> (accessed on 1 February 2022).
83. Shoham, Y. *Multiagent Systems—Algorithmic, Game-Theoretic, and Logical Foundations*; ITU-R Report M.2334-0 (11/2014). Stanford University and Kevin Leyton-Brown, University of British Columbia: Vancouver, BC, Canada, 2010; Available online: <http://www.masfoundations.org> (accessed on 3 February 2022).
84. Xiong, Z.; Zhang, Y.; Niyato, D.; Deng, R.; Wang, P.; Wang, L.C. Deep Reinforcement Learning for Mobile 5G and Beyond: Fundamentals, Applications, and Challenges. *IEEE Veh. Technol. Mag.* **2019**, *14*, 44–52. [[CrossRef](#)]
85. Tanveer, J.; Haider, A.; Ali, R.; Kim, A. An Overview of Reinforcement Learning Algorithms for Handover Management in 5G Ultra-Dense Small Cell Networks. *Appl. Sci.* **2022**, *12*, 426. [[CrossRef](#)]
86. Mismar, F.B.; Evans, B.L.; Alkhateeb, A. Deep Reinforcement Learning for 5G Networks: Joint Beamforming, Power Control, and Interference Coordination. *IEEE Trans. Commun.* **2020**, *68*, 1581–1592. [[CrossRef](#)]
87. Raza, M.R.; Natalino, C.; Öhlen, P.; Wosinska, L.; Monti, P. Reinforcement Learning for Slicing in a 5G Flexible RAN. *J. Lightwave Technol.* **2019**, *37*, 5161–5169. [[CrossRef](#)]
88. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning Series)*, 2nd ed.; The MIT Press: Cambridge, MA, USA, 2020.
89. Galindo-Serrano, A.; Giupponi, L.; Blasco, P.; Dohler, M. Learning from experts in cognitive radio networks: The docitive paradigm. In Proceedings of the 2010 Fifth International Conference on Cognitive Radio Oriented Wireless Networks and Communications, Cannes, France, 9–11 June 2010; pp. 1–6.
90. Cisco ACI Adopted for NTT DOCOMO's Network Functions Virtualization (NFV). 2016. Available online: <https://newsroom.cisco.com/press-release-content?articleId=1753087> (accessed on 3 February 2022).
91. Cloud RAN Architecture for 5G, White Paper. Available online: [http://www.hit.bme.hu/~{jakab/edu/litr/5G/WhitePaper\\_C-RAN\\_for\\_5G\\_Telefonica\\_Ericsson.PDF](http://www.hit.bme.hu/~{jakab/edu/litr/5G/WhitePaper_C-RAN_for_5G_Telefonica_Ericsson.PDF) (accessed on 3 February 2022).
92. The Benefits of NFV are Unfolding in 5G, White Paper by ERICSSON. Available online: <https://www.ericsson.com/en/blog/2019/10/the-benefits-of-nfv-are-unfolding-in-5g> (accessed on 3 February 2022).
93. How Cellwise is Reinventing Innovation in Telecom with SON. Available online: <https://telecomdrive.com/how-cellwise-is-reinventing-innovation-in-telecom-with-son/> (accessed on 3 February 2022).
94. Westerberg, E. 4G/5G RAN Architecture: How a Split Can Make the Difference, Ericsson Technology Review (Online), July 2016. Available online: <https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/4g5g-ran-architecture-how-a-split-can-make-the-difference> (accessed on 3 February 2022).
95. 5GPPP. View on 5G Architecture, v3.0. White Paper. June 2019. Available online: <https://5g-ppp.eu/white-papers/> (accessed on 3 February 2022).
96. ETSI GS NFV 002, V1.2.1. Network Functions Virtualization (NFV), Architectural Framework. 2014-12. Available online: [https://www.etsi.org/deliver/etsi\\_gs/nfv/001\\_099/002/01.02.01\\_60/gs\\_nfv002v010201p.pdf](https://www.etsi.org/deliver/etsi_gs/nfv/001_099/002/01.02.01_60/gs_nfv002v010201p.pdf) (accessed on 3 February 2022).
97. China Mobile. *C-RAN, The Road Towards Green RAN*; White Paper Version 3.0; IEEE: Piscataway, NJ, USA, 2010.
98. Checko, A.; Christiansen, H.L.; Yan, Y.; Scolari, L.; Kardaras, G.; Berger, M.S.; Dittmann, L. Cloud RAN for Mobile Networks—A Technology Overview. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 405–426. [[CrossRef](#)]
99. NGMN (Next Generation Mobile Networks). *Version 1.0, Further Study on Critical C-RAN Technologies*; Alliance: Hong Kong, China, 2015.
100. Hossain, F.; Mahin, A.U.; Debnath, T.; Mosharraf, F.B.; Islam, K.Z. Recent Research in Cloud Radio Access Network (C-RAN) for 5G Cellular Systems—A Survey. *J. Netw. Comput. Appl.* **2019**, *139*, 31–48. [[CrossRef](#)]
101. Rodoshi, R.T.; Kim, T.; Choi, W. Resource Management in Cloud Radio Access Network: Conventional and New Approaches. *Sensors* **2020**, *20*, 2708. [[CrossRef](#)]
102. Goodfellow, I.; Bengio, Y.; Courville, A. Deep Learning. 2016. Available online: <https://www.deeplearningbook.org> (accessed on 1 February 2022).
103. Ouyang, Z.; Li, L.; Su, W.L.; Lin, Z. APP-SON: Application characteristics-driven SON to optimize 4G/5G network performance and quality of experience. In Proceedings of the 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, 11–14 December 2017; pp. 1514–1523.
104. VIAVINITRO. GEOperformance. Overview. 2020. Available online: <https://www.viavisolutions.com/de-de/literature/nitro-geo-geperformance-overview-brochure-en.pdf> (accessed on 3 February 2022).
105. Björnson, E.; Sanguinetti, L.; Wymeersch, H.; Hoydis, J.; Marzetta, T.L. Massive MIMO Is a Reality—What Is Next? Five Promising Research Directions for Antenna Arrays. *Digit. Signal Process.* **2019**, *94*, 3–20. [[CrossRef](#)]
106. The Benefits of Self-Organizing Backhaul Networks, White Paper by ERICSSON. Available online: <https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/the-benefits-of-self-organizing-backhaul-networks> (accessed on 3 February 2022).
107. Jaber, M.; Imran, M.A.; Tafazolli, R.; Tukmanov, A. 5G Backhaul Challenges and Emerging Research Directions: A Survey. *IEEE Access* **2016**, *4*, 1743–1766. [[CrossRef](#)]

108. Mobile Backhaul Options. White Paper by GSMA. Available online: <https://www.gsma.com/spectrum/resources/mobile-backhaul-options/> (accessed on 3 February 2022).
109. Bajzik, L.; Kárász, T.; Vincze, Z.; Vulkán, C.S.; Ben Ameer, W.; Altman, Z.; Diascorn, V. SON for Mobile Backhaul. In Proceedings of the IEEE Wireless Communications and Networking Conference Workshops (WCNCW), Barcelona, Spain, 15–18 April 2018.
110. Prazeres, C.; Serrano, M. SOFT-IoT: Self-Organizing FOG of Things. In Proceedings of the 2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA), Crans-Montana, Switzerland, 23–25 March 2016; IEEE: Manhattan, NY, USA, 2016; pp. 803–808.
111. Domaschka, J.; Berger, C.; Reiser, H.; Eichhammer, P.; Griesinger, F.; Pietron, J.; Tichy, M.; Hauck, F.; Habiger, G. SORRIR: A Resilient Self-Organizing Middleware for IoT Applications [Position Paper]. In Proceedings of the Middleware 19: 20th International Middleware Conference, Davis, CA, USA, 9–13 December 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 13–16.
112. Azari, A.; Cavdar, C. Self-Organized Low-Power IoT Networks: A Distributed Learning Approach. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, 9–13 December 2018; pp. 1–7.
113. Lucas-Estan, M.C.; Gozalvez, J. Load Balancing for Reliable Self-Organizing Industrial IoT Networks. *IEEE Trans. Ind. Inform.* **2019**, *15*, 5052–5063. [[CrossRef](#)]
114. Holtrup, G.; Lacube, W.; Percia, D.D.; Mermoud, A.; Bovet, G.; Lenders, V. 5G System Security Analysis. *arXiv* **2021**, arXiv:2108.08700.
115. Ji, H.; Park, S.; Yeo, J.; Kim, Y.; Lee, J.; Shim, B. Introduction to Ultra-Reliable and Low Latency Communications in 5G. CoRR. 2017. arXiv:1704.05565. Available online: <http://arxiv.org/abs/1704.05565> (accessed on 3 February 2022).
116. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [[CrossRef](#)]
117. Asadi, A.; Wang, Q.; Mancuso, V. A Survey on Device-to-Device Communication in Cellular Networks. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1801–1819. [[CrossRef](#)]
118. ETSI TR 102 725, V1.1.1, Machine-to-Machine Communications (M2M), Definitions; ETSI. 2013. Available online: [https://www.etsi.org/deliver/etsi\\_tr/102700\\_102799/102725/01.01.01\\_60/tr\\_102725v010101p.pdf](https://www.etsi.org/deliver/etsi_tr/102700_102799/102725/01.01.01_60/tr_102725v010101p.pdf) (accessed on 3 February 2022).
119. Taleb, T.; Kunz, A. Machine Type Communications in 3GPP Networks: Potential, Challenges, and Solutions. *IEEE Commun. Mag.* **2012**, *50*, 178–184. [[CrossRef](#)]
120. Upcoming Special Issues of the ITU Journal Look towards 6G. 2021. Available online: <https://www.itu.int/hub/2021/09/upcoming-special-issues-of-the-itu-journal-look-towards-6g/> (accessed on 1 February 2022).
121. Why Are We Talking About 6G Already? 2021. Available online: <https://news.vmware.com/technologies/what-is-6g> (accessed on 3 February 2022).
122. Self-Organizing Networks in the 6G Era: State-of-the-Art, Opportunities, Challenges, and Future Trends. *arXiv* **2021**, arXiv:2112.09769.
123. Dryjański, M.; Kułacz, Ł.; Kliks, A. Toward Modular and Flexible Open RAN Implementations in 6G Networks: Traffic Steering Use Case and O-RAN xApps. *Sensors* **2021**, *21*, 8173. [[CrossRef](#)]
124. Xie, X.; Rong, B.; Kadoch, M. *6G Wireless Communications and Mobile Networking*; Bentham Books: Sharjah, United Arab Emirates, 2021.

MDPI  
St. Alban-Anlage 66  
4052 Basel  
Switzerland  
Tel. +41 61 683 77 34  
Fax +41 61 302 89 18  
[www.mdpi.com](http://www.mdpi.com)

*Future Internet* Editorial Office  
E-mail: [futureinternet@mdpi.com](mailto:futureinternet@mdpi.com)  
[www.mdpi.com/journal/futureinternet](http://www.mdpi.com/journal/futureinternet)





MDPI  
St. Alban-Anlage 66  
4052 Basel  
Switzerland

Tel: +41 61 683 77 34

[www.mdpi.com](http://www.mdpi.com)



ISBN 978-3-0365-6807-2