UNICO I+D Project 6G-INTEGRATION-3 (TSI-063000-2021-127)

6G-INTEGRATION-3

Enhanced innovations for the NTN integration with 3GPP networks

Abstract

This document provides a summary of examples for enhanced innovations in the NTN integration with 3GPP networks. The main innovations overviewed have to do with the applicability of modern AI/ML algorithms to help modeling, solving and optimizing different aspects of NTNs built with both terrestrial equipment and on-air and space equipment. Main challenges in these types of networks have to do with dealing with high latency, the errors due to radiation and transmission impairments,

Financiado por la Unión Europea NextGenerationEU

GOBIERNO DE ESPAÑA MINISTERIO DE ASUNTOS ECONÓMICOS Y TRANSFORMACIÓN DIGITAL SECRETARÍA DE ESTADO DE TELECOMUNICACIONES E INFRAESTRUCTURAS DIGITALES

Plan de Recuperación, Transformación y Resiliencia

Doppler shifts, mobility management, handovers between terrestrial and non-terrestrial networks, channel reliability. Multi-access Edge Computing (MEC) can provide caching and computing resources on board to provide near real-time applications to minimise latency. Also intelligent algorithms based on Reinforcement Learning and other AI/ML strategies can be used to optimise network performance from multiple sides: resource optimization, optimal routing, network slicing and mobility management. This document provides an overview of such strategies and algorithms toward a real integration of both terrestrial and non-terrestrial networks with current 5G deployments and emerging 6G networks.

Document properties

Document number	Deliverable E6 for project 6G-INTEGRATION-3
Document title	Enhanced innovations for the NTN integration with 3GPP networks
Document responsible	José Alberto Hernández
Document editor	José Alberto Hernández
Editorial team	Alfonso Sánchez-Macián, Fernando Lledó, David Larrabeiti
Target dissemination level	Public
Status of the document	Final
Version	1.0
Delivery date	31st Dec 2023
Actual delivery date	18th Jan 2024

Production properties

Reviewers	Alfonso Sánchez-Macián, Fernando Lledó, David Larrabeiti

Disclaimer

This document has been produced in the context of the 6G-INTEGRATION-3 Project (grant number TSI-063000-2021-127). The research leading to these results has received funding from the Spanish Ministry of Economic Affairs and Digital Transformation and the European Union-NextGenerationEU through the UNICO 5G I+D programme.

Contents

Resumen Ejecutivo

Este documento proporciona un resumen con ejemplos de innovaciones para la integración de NTN con redes 3GPP. Las principales innovaciones incluidas tienen que ver con la aplicabilidad de algoritmos modernos de IA/ML para ayudar a modelar, resolver y optimizar diferentes aspectos de las NTNs construidas tanto con equipos terrestres como con equipos aéreos y espaciales. Los principales desafíos en estos tipos de redes tienen que ver con las formas de lidiar con alta latencia, los errores debidos a la radiación y las deficiencias de transmisión, los desplazamientos Doppler, la gestión de la movilidad, los handovers entre redes terrestres y no terrestres, y la fiabilidad del canal de comunicaciones. El Multi-access Edge Computing (MEC) puede proporcionar recursos de almacenamiento y computación a bordo para ofrecer aplicaciones casi en tiempo real y minimizar la latencia. Además, algoritmos inteligentes basados en Aprendizaje por Refuerzo y otras estrategias de IA/ML se pueden utilizar para optimizar el rendimiento de la red desde múltiples frentes: optimización de recursos, enrutamiento óptimo, segmentación de red y gestión de la movilidad. Este documento proporciona una visión general de tales estrategias y algoritmos hacia una integración real de redes terrestres y no terrestres con las implementaciones actuales de 5G y las emergentes redes 6G.

Financiado por la Unión Europea NextGenerationEU
GOBIERNO DE ESPAÑA MINISTERIO DE ASUNTOS ECONÓMICOS Y TRANSFORMACIÓN DIGITAL SECRETARÍA DE ESTADO DE TELECOMUNICACIONES E INFRAESTRUCTURAS DIGITALES
Plan de Recuperación, Transformación y Resiliencia
UNICO I+D

Executive Summary

This document provides a summary of examples for enhanced innovations in the NTN integration with 3GPP networks. The main innovations overviewed have to do with the applicability of modern AI/ML algorithms to help modeling, solving and optimizing different aspects of NTNs built with both terrestrial equipment and on-air and space equipment. Main challenges in these types of networks have to do with dealing with high latency, the errors due to radiation and transmission impairments, Doppler shifts, mobility management, handovers between terrestrial and non-terrestrial networks, channel reliability. Multi-access Edge Computing (MEC) can provide caching and computing resources on board to provide near real-time applications to minimise latency. Also intelligent algorithms based on Reinforcement Learning and other AI/ML strategies can be used to optimise network performance from multiple sides: resource optimization, optimal routing, network slicing and mobility management. This document provides an overview of such strategies and algorithms toward a real integration of both terrestrial and non-terrestrial networks with current 5G deployments and emerging 6G networks.

1. Introduction

As we stand on the cusp of a fully interconnected world, the advent of 5G technology brings with it the promise of unprecedented communication capabilities. A pivotal element in realizing this promise is the integration of non-terrestrial networks (NTNs) into the 5G architecture. NTNs encompass a diverse array of communication platforms, including satellites in various orbits – from geostationary (GEO) to medium earth orbit (MEO) and low earth orbit (LEO) – as well as airborne networks like high-altitude platform stations (HAPS), see FIGURE 1.

The role of NTNs in 5G is transformative [Msadaa22]. They are set to extend connectivity to the most remote and rural areas, overcoming the geographical and economic challenges that terrestrial infrastructure faces. With the inherent high reliability and broad coverage of NTNs, 5G services can be uniformly distributed, ensuring that no area is left behind in the digital divide. Furthermore, NTNs are crucial in providing enhanced support for critical communication services that require high availability, such as disaster response, where terrestrial networks might be compromised.

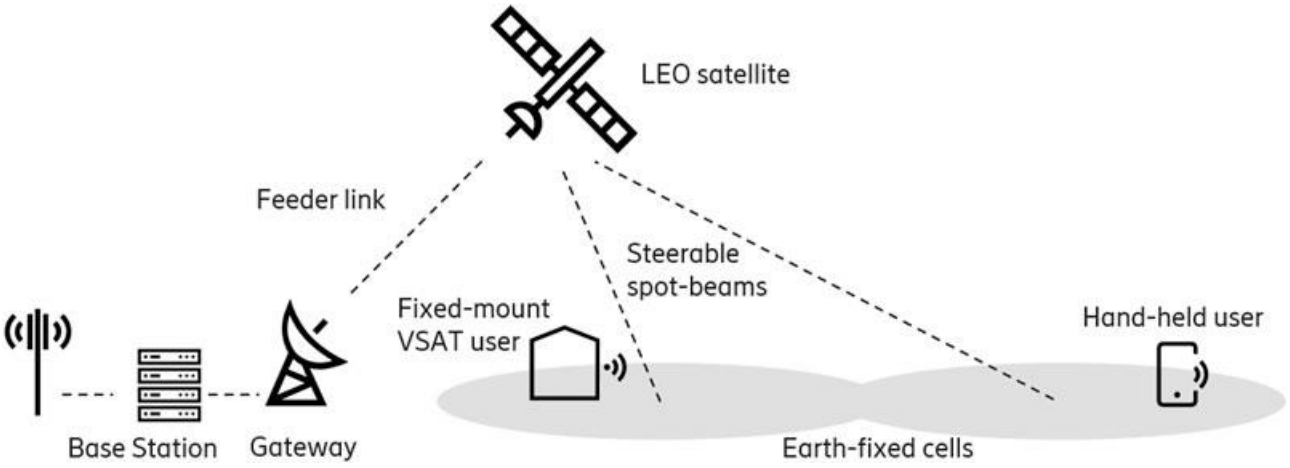

FIGURE 1 EXAMPLE OF A NON-TERRESTRIAL NETWORK

One of the key advantages of NTNs is their ability to deliver 5G's hallmark ultra-reliable low-latency communications (URLLC) to every corner of the globe. This is instrumental for applications like telemedicine, automated transportation, and IoT networks, where a delay of even milliseconds could be detrimental. Moreover, the integration of NTNs enables the seamless mobility of 5G services, maintaining consistent connectivity for users on the move, whether they are flying across continents or sailing the oceans.

However, the integration of non-terrestrial networks (NTNs) with 5G technology presents a unique set of challenges that stem from the intrinsic characteristics of space-based and airborne communication systems. Addressing the challenges of the Doppler effect, latency, mobility management, and handovers is crucial for the successful integration of NTNs into the 5G ecosystem [Geraci23, Vanelli-Coralli20].

Concerning the Doppler effect, this is particularly harmful in NTNs due to the high relative speed of satellites, especially in LEO orbits, or the mobility of airborne platforms like HAPS. The Doppler effect

can cause a shift in the frequency of the signal received by a moving object, leading to potential issues in signal demodulation and synchronization. In a 5G context, where high-frequency bands are in use, the impact of the Doppler effect can be even more significant, potentially affecting the reliability of the communication link. Mitigating these effects requires advanced signal processing techniques and adaptive communication protocols that can dynamically compensate for the frequency variations, along with AI/ML techniques.

Another challenge is latency. While NTNs can reduce latency compared to GEO satellites due to their closer proximity to the Earth's surface, they still introduce greater latency compared to terrestrial networks, particularly for LEO-based systems which require signals to travel significant distances to and from the satellites. This latency can be a barrier to real-time applications that are latency-sensitive, which are a cornerstone of 5G services, such as autonomous driving and tactile internet applications. Strategies to mitigate latency involve optimizing the satellite constellations, reducing processing times, and employing edge computing techniques to process data closer to end-users.

Mobility management is also a complex challenge in integrating NTNs with 5G. As users move, the network must seamlessly manage connections across rapidly changing network topologies. This is particularly challenging with NTNs, where satellites or HAPS are constantly moving, requiring frequent handovers between different non-terrestrial network elements or between terrestrial and non-terrestrial components. Efficient mobility management protocols must be developed to ensure seamless service continuity without interruption, which is essential for a consistent user experience.

Handover in an NTN context involves transferring the user's connection from one satellite to another or between NTNs and terrestrial networks. This process is complicated by the high velocity of LEO satellites and the potential for a large number of handovers. Each handover must be executed flawlessly to maintain the Quality of Service (QoS) and avoid data loss or connection drops. Advanced predictive algorithms and real-time network analytics are required to anticipate the need for handover and prepare the network resources accordingly, minimizing the handover failure rate and ensuring a smooth transition.

This document evaluates the uses of AI/ML algorithms to address these set of challenges of NTN-5G integration, and others too. In particular, Section 2 provides a quick overview of the physical layer of LTE and 5G along with possible architectural solutions of Non-Terrestrial Networks to be deployed for 5G/6G support. Section 3 reviews critical aspects of hardware reliability for equipment to be placed on board in satellites, and resiliency against errors due to space radiation and other onboard impairments. Section 4 reviews existing AI/ML algorithms in the literature and their possible applications in NTNs, including resource optimisation, network routing and effective caching for edge computing. Section 5 delves into 3GPP EdgeApp, an standard solution for incorporating Mobile Edge Computing (MEC) applications directly in the remote units of 5G architectures. Section 6 provides and evaluation of Round-Trip Time (RTT) latency in Low-Earth Orbit satellites and their comparison against fiber transmission delays, using both theoretical and simulation tools. Finally, Section 7 concludes this work with a summary of its main contributions.

2. A brief overview of 5G and LTE physical layer

2.1 An overview of the physical layer of LTE and 5G for terrestrial networks

The physical layer is fundamental in LTE, as it directly impacts the efficiency, reliability, and speed of the network. LTE is based on Orthogonal Frequency-Division Multiplexing (OFDM), which is a digital multi-carrier modulation method used extensively in wireless communication, including LTE. It works by dividing a high data rate modulating stream into multiple lower data rate streams [Huang16]. Each of these streams is then transmitted on a different frequency carrier. These carriers are orthogonal to each other, meaning their cross-correlation is zero, which minimizes interference (FIGURE 2).

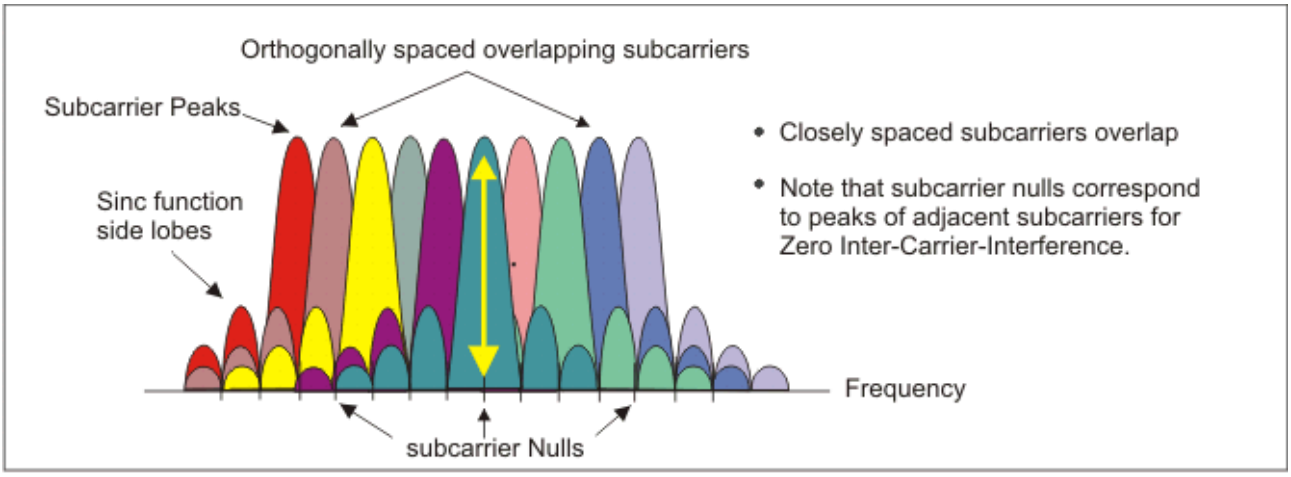

FIGURE 2 OFDM PRINCIPLES OF OPERATION

OFDM is chosen for LTE due to several compelling advantages:

- Robustness to Multipath Fading: OFDM combats multipath fading effectively, a common issue in wireless communication where signals arrive at the receiver via different paths.
- Spectral Efficiency: It makes efficient use of the spectrum by closely spacing the subcarriers.
- Flexibility: OFDM allows for flexible allocation of resources depending on the user's need and channel conditions.
- Simplification of Equalization: The use of Fast Fourier Transform (FFT) in OFDM simplifies the equalization process required to counteract channel-induced distortions.

LTE supports various modulation schemes, each with different spectral efficiency and Signal-to-Interference-Noise Ratio (SNR) requirements, ranging from simple QPSK to 64 QAM:

- QPSK (Quadrature Phase Shift Keying) is the simplest form used in LTE. It is robust but has lower spectral efficiency, it maps two bits per symbol and is used in poor channel conditions.

- 16-QAM (Quadrature Amplitude Modulation): A more spectrally efficient modulation, 16-QAM maps four bits per symbol. It requires a better SNR than QPSK.
- 64-QAM: This format offers even higher spectral efficiency by mapping six bits per symbol. However, it demands a high SNR, making it suitable for good channel conditions (FIGURE 3).

Table 1: Modulation formats at typical minimum SINR required for operation

Modulation Format	Minimum SINR (approximate, in dB)
QPSK	-2 to 5 dB
16-QAM	10 to 13 dB
64-QAM	16 to 20 dB
256-QAM	22 to 25 dB

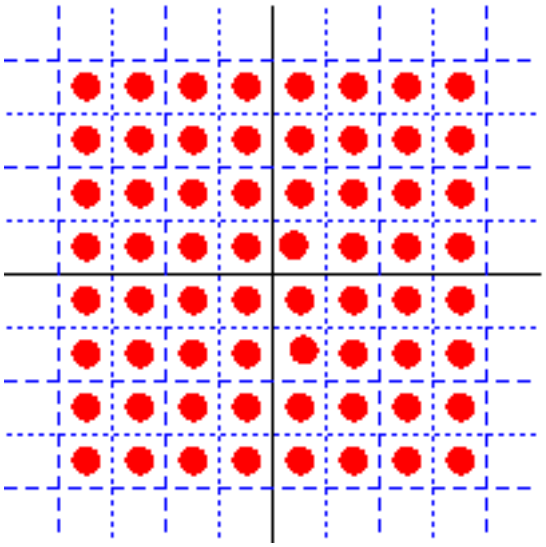

FIGURE 3 EXAMPLE OF 64 QAM CONSTELLATION

In 5G, modulation format can go up to 1024 QAM which implies 10 bits per symbol but requires very good SINR levels.

Concerning the frame structure in LTE, this uses a radio frame structure that is divided into subframes and further into slots. A standard LTE radio frame is 10 ms long, divided into 10 subframes, each 1 ms in duration. Each subframe consists of two consecutive slots of 0.5 ms, and each slot carries 6 or 7 OFDM symbols (FIGURE 4).

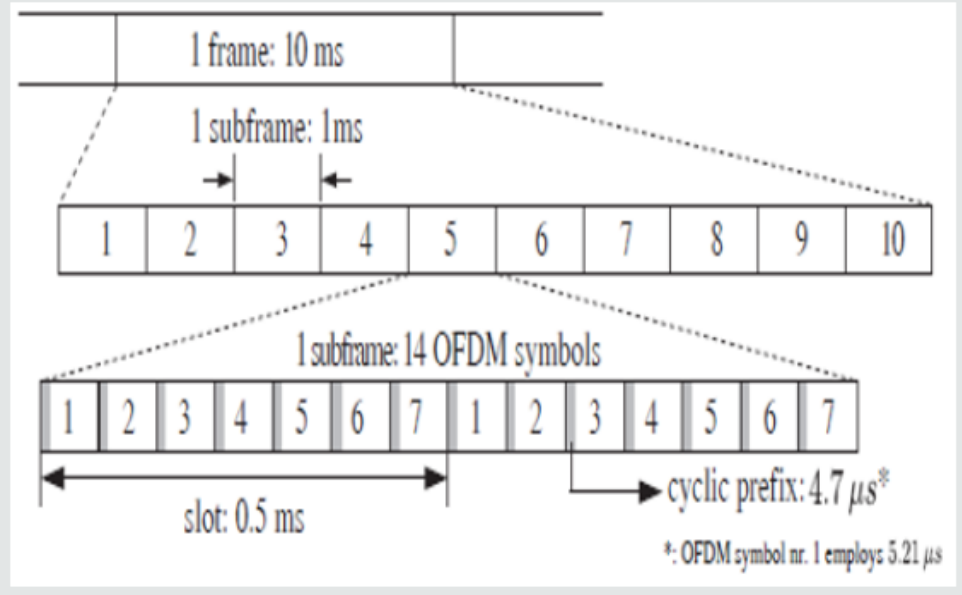

FIGURE 4 LTE FRAME FORMAT

The key element in the LTE frame structure is the Physical Resource Block (PRB), see FIGURE 5. PRBs are the smallest units of resource allocation in the LTE physical layer [Oliva16]. A PRB consists of 12 consecutive subcarriers in frequency for one slot in time. The number of PRBs in a bandwidth varies according to the system bandwidth (ranging from 6 PRBs in a 1.4 MHz system to 100 PRBs in a 20 MHz system). PRBs are crucial for resource allocation and scheduling, as they are dynamically assigned to users based on their data needs and channel conditions.

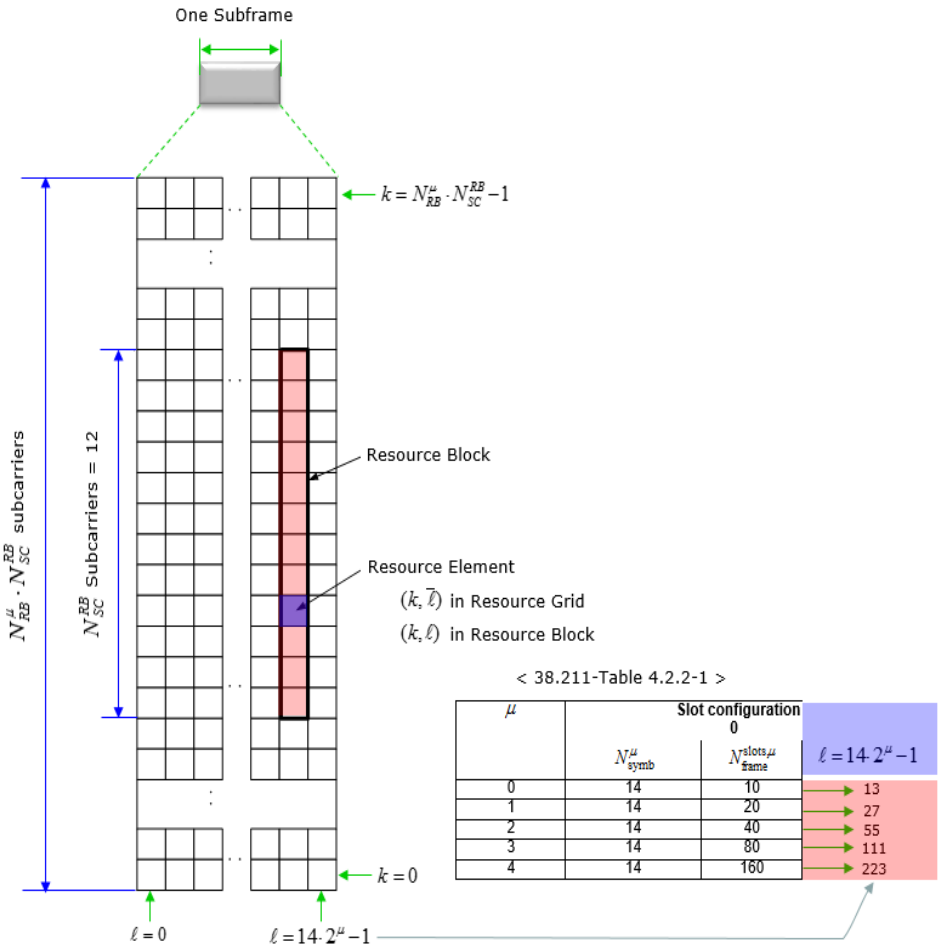

FIGURE 5 PHYSICAL RESOURCE BLOCKS IN LTE

LTE and 5G employs Hybrid Automatic Repeat reQuest (HARQ), which is a protocol commonly used in wireless communication to improve data transmission reliability. In the context of 5G, HARQ becomes particularly important due to the high data rates and the need for efficient error correction [Khosravirand16].

Essentially, HARQ is a combination of Automatic Repeat reQuest (ARQ) and Forward Error Correction (FEC), see FIGURE 6. ARQ is a protocol for error detection and retransmission, while FEC adds extra data bits (redundancy) to the original signal to help detect and correct errors without the need for retransmission. When data is transmitted, it is encoded with error correction codes. The receiver attempts to decode the message. If the message is successfully decoded, it sends an acknowledgment (ACK) back to the transmitter. If the receiver cannot decode the message correctly (due to errors like interference or weak signal), it sends a negative acknowledgment (NACK) back to the transmitter. Upon receiving a NACK, instead of retransmitting the original data, the transmitter sends additional error correction information. This is different from traditional ARQ, where the entire message would be resent. The receiver combines this new information with the previously received erroneous data to attempt a successful decoding.

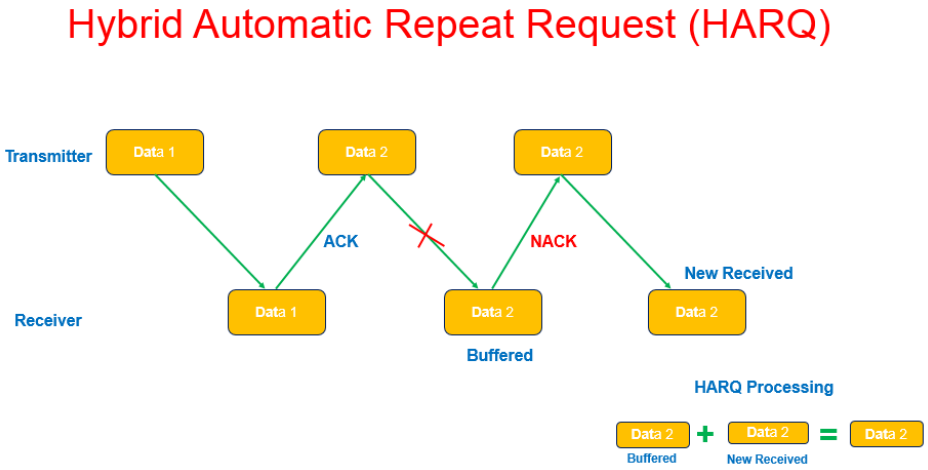

FIGURE 6 EXAMPLE OF HARQ OPERATION

This process is more efficient than traditional ARQ because it requires less retransmission. The use of FEC in combination with ARQ allows for more robust error correction, especially in environments where signal quality can vary significantly.

In 5G networks, with their high data rate requirements and diverse service scenarios (like enhanced mobile broadband, ultra-reliable low latency communications, and massive machine-type communications), the efficiency and reliability of data transmission are critical. HARQ helps achieve this by minimizing the need for retransmissions and enabling more reliable communication, even in challenging conditions.

2.2 Cloud RAN and Coordinated Multi-Point (CoMP)

Cloud RAN and Coordinated Multipoint, two pivotal concepts in modern wireless communication, particularly relevant in the context of 5G and beyond. Cloud Radio Access Network (Cloud RAN or C-RAN) is a transformative architecture in wireless networking that shifts the traditional RAN design towards a centralized, cloud-based model. This paradigm is increasingly vital in the era of 5G, where network flexibility, scalability, and efficiency are paramount. The centralization in Cloud RAN facilitates easier implementation of CoMP by pooling resources and processing capabilities. Cloud RAN's flexible architecture allows for more dynamic and efficient CoMP operations, adapting to real-time network conditions.

In a traditional RAN, each cell site has its own base station, with the hardware and software required for signal processing located on-site. Cloud RAN, by contrast, centralizes these resources. The essential components of Cloud RAN include:

- Remote Radio Heads (RRHs): Located at the cell site, RRHs are responsible for transmitting and receiving radio frequency signals.
- Baseband Units (BBUs): In Cloud RAN, BBUs are centralized in a few locations or data centers, performing the baseband processing for multiple RRHs.
- Front Haul Network: This is the high-speed, low-latency communication link connecting RRHs with centralized BBUs.

There are multiple advantages of Cloud RAN, including scalability, cost-effectiveness, energy-efficiency and flexibility/agility. By centralizing resources, Cloud RAN can dynamically allocate processing power where it's needed, efficiently handling varying traffic loads (scalability). In addition, C-RAN reduces the need for expensive hardware at each site, lowering capital and operational expenses (cost-effectiveness). Further, centralization allows for better energy management and lower power consumption. Finally, C-RAN supports the deployment of new services and technologies without significant hardware changes (flexibility).

However, in order to be properly deployed, C-RAN poses a number of challenges including the need for deployment a high-speed ultra-low latency fronthaul networks for the support of radio over fiber traffic, along with the security and reliability requirements of the centralised processing included in the C-RAN solution.

Nevertheless, centralised radio functions will enable the use of Coordinated Multipoint (CoMP), which is a set of techniques designed to enhance the performance of the radio bandwidth, especially at cell edges where interference is a significant issue and signal quality typically degrades [Irram20].

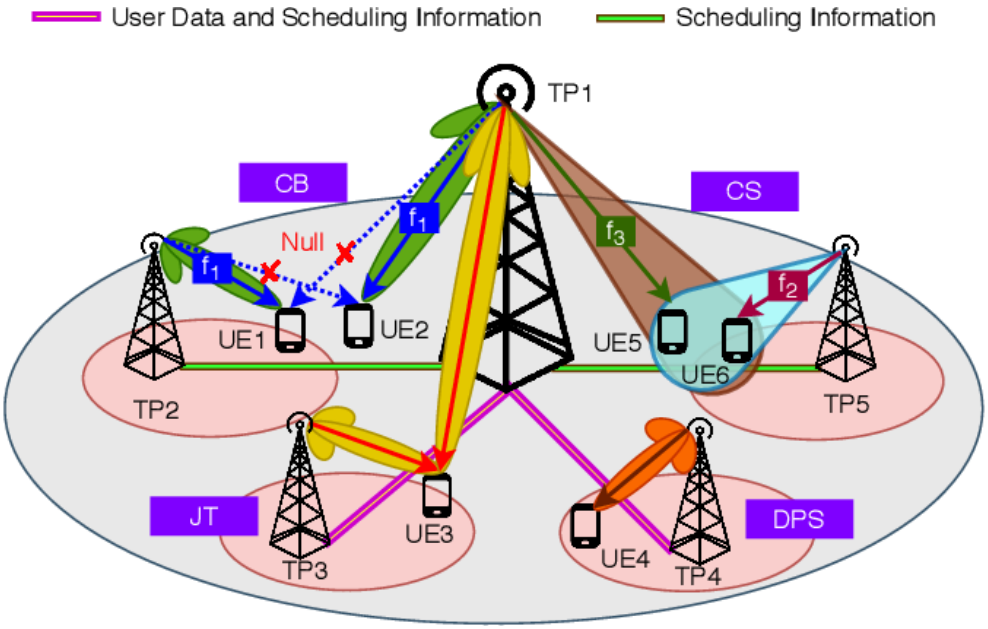

FIGURE 7 USE CASE SCENARIO OF COORDINATED MULTIPOINT [SOLAIJA21]

Indeed, CoMP is particularly crucial in dense, urban environments and for applications demanding high reliability and throughput, such as in 5G networks [Solaija21]. CoMP involves the coordination of multiple base stations or cell sites to serve a single user or multiple users simultaneously. This coordination can take various forms, the most important ones are (FIGURE 7):

- Joint Transmission and Joint Reception: Multiple base stations transmit the same data to a user, thereby improving signal strength and reliability.
- Coordinated Scheduling/Beamforming: Base stations coordinate their scheduling decisions and beamforming strategies to reduce interference and optimize network capacity.
- Dynamic Cell Selection: Users dynamically connect to the best cell site based on signal quality, load, and other factors.

With all these techniques operating in a centralised location, several benefits are expected, namely Improved Edge Performance, Increased Throughput and Spectral Efficiency, ultimately leading to Enhanced User Experience.

The integration of these technologies is pivotal in the evolution towards more advanced wireless networks, including 5G and beyond. They support key 5G goals like high throughput, low latency, massive connectivity, and enhanced mobility management.

In urban areas with high user density, the combination of Cloud RAN and CoMP can efficiently manage network traffic, reducing congestion and enhancing user experience (enhanced Mobile BroadBand services, eMBB) [Campana23]. For applications requiring high reliability and low latency, such as remote surgery or autonomous vehicles, the integration ensures robust and responsive network performance (Ultra Reliable Low-Latency Communication services, URLLC). In scenarios with many connected devices, such as IoT networks, this integration supports scalability and efficient resource utilization (Massive Machine Type Communications services, mMTC).

C-RAN will also enable a number of technological innovations, including:

- Artificial Intelligence and Machine Learning: AI/ML can play a crucial role in optimizing Cloud RAN and CoMP operations, enabling predictive analytics, dynamic resource allocation, and intelligent network management.
- Advanced Beamforming: Leveraging advanced beamforming techniques in this integrated setup enhances signal focus and interference management, crucial for maintaining high-quality connections.
- Network Slicing: This integration is essential for implementing network slicing in 5G, allowing the creation of multiple virtual networks with distinct characteristics to cater to diverse application requirements.

Analog radio signals are typically packetized and encapsulated for their transport over the fronthaul network using the Radio over Ethernet frame format. Radio over Ethernet (RoE) is a method of transporting radio signals over standard Ethernet networks on attempts to leverage the ubiquity, flexibility, and cost-effectiveness of Ethernet technology for the fronthaul part of the network. Such an encapsulation involves adding headers and other control information (like VLANs for traffic priorization) to the radio signal data so that it can be properly routed and processed within the Ethernet network.

2.3 5G and envisioned 6G services adapted for Non-Terrestrial Networks

Non-Terrestrial Networks (NTNs), comprising of satellites and High Altitude Platform Stations (HAPS), present unique challenges when adapting traditional 5G services. These challenges stem from inherent satellite characteristics such as higher latency, path loss, and Doppler shift. Classical 5G use cases—enhanced Mobile Broadband (eMMB), Massive Machine Type Communications (mMTC), and Ultra-Reliable and Low Latency Communications (uRLLC)—require adjustments to meet these challenges (FIGURE 8) [Khan22]. For instance, the International Mobile Telecommunications-

2020 (IMT-2020) standards for satellite radio interfaces suggest different performance metrics for NTNs, including peak data rates of 70 Mbit/s in downlink and 2 Mbit/s in uplink, user plane latency of 650 ms, and mobility support for speeds up to 1200 km/h. These specifications, while diverging from typical terrestrial networks, are crucial for ensuring efficient and reliable satellite communication.

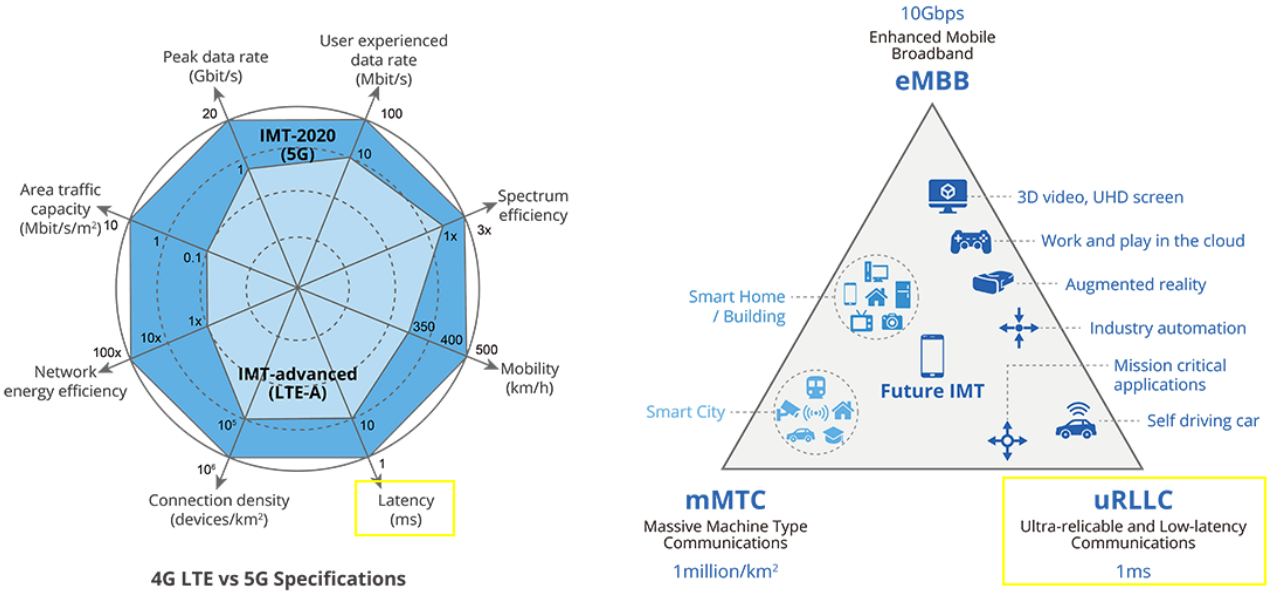

FIGURE 8 5G SERVICES ENVISIONED BY IMT-2020

The unique attributes of NTNs necessitate the development of specialized services, tailored to their capabilities and limitations. These services can be broadly categorized into three areas: service continuity, throughput enhancement, and ubiquitous connectivity.

1. Service Continuity: NTNs are invaluable in ensuring continuous connectivity, especially in mobile platforms like ships, airplanes, and high-speed trains. These platforms, often in open spaces and at high speeds, benefit from the coverage and speed capabilities of LEO satellites and HAPS. For instance, each passenger can be connected at rates as high as 50 Mbit/s, addressing the intermittent connectivity issues prevalent in such scenarios.
2. Ubiquitous Connectivity: NTNs play a crucial role in providing internet access to underserved or isolated areas. This includes enabling eMMB backhauling for remote access points, supporting mMTC for sensors in remote areas, and facilitating critical infrastructure monitoring. In these scenarios, the choice of NTEs—whether HAPS, LEO satellites, or UAVs—depends on the specific requirements, such as area coverage, data rate needs, and the criticality of the monitored infrastructure.
3. Throughput Enhancement: In regions where, terrestrial networks are present but overburdened, NTNs can supplement network capacity, thus enhancing user experience. This dual connectivity, managed by both terrestrial and non-terrestrial elements, ensures efficient data rate distribution and reliable communication, particularly during peak usage periods or in congested areas.

Apart from these, NTNs offer vital solutions in situations like disaster relief, where rapid deployment and high data rate capabilities can aid in rescue operations and emergency communications. They also provide backup connectivity for remote operations in areas lacking terrestrial network coverage. Additionally, NTNs facilitate broadcasting and multicasting services, leveraging their wide coverage to deliver high-definition content in areas with limited terrestrial connectivity.

Thus, the adaptation of 5G services to NTNs is not just a matter of replicating terrestrial capabilities in a non-terrestrial setting. It involves a careful reevaluation and modification of these services to align with the unique characteristics and potential of NTNs. By doing so, we can extend the benefits of 5G technology to environments and applications previously beyond its reach, ensuring broader connectivity and enhanced service scenarios.

2.4 Architectural solutions for deploying non-terrestrial networks with 5G support

Different architectural solutions for NTNs for supporting the previous services have been proposed in the literature. These architectures are primarily differentiated by the location and functionality of the gNB (gNodeB) components, leading to three basic types: transparent, regenerative, and split architectures.

1. Transparent NTN Architecture: In the transparent architecture, the gNB is situated on the ground, positioned after the NTN ground station. Here, the Non-Terrestrial Entity (NTE) acts simply as a conduit, transparently relaying signals from the User Equipment (UE) to the NTN gateway without on-board processing, except for possible frequency translation. This setup implies that the entire protocol stack, from the physical layer (PHY) to the service data adaptation protocol (SDAP), is managed on the ground. In the control plane (CP), protocols from PHY to Radio Resource Control (RRC) are closed at the gNB. This architecture is particularly useful for straightforward relay operations where minimal on-board processing is required, hence reducing the complexity and cost of the satellite system.
2. Regenerative NTN Architecture: In contrast, the regenerative architecture involves mounting the gNB on-board the NTE. This configuration enhances NTN performance by enabling on-board processing of signals. The satellite radio interface (SRI) in the feeder link is a transport link used for transmitting both user and control data from the NTE to the ground-based NTN gateway. The key advantage of this architecture is the ability to process and regenerate signals on-board, leading to potentially better signal quality and more efficient use of the satellite's resources. The protocol stack for regenerative NTN sees the access stratum protocols closing on-board, with the NTN gateway remaining transparent in the process.
3. On-Board Distributed Unit (DU) NTN Architecture: The split architecture, or the on-board DU NTN architecture, represents a middle ground between the transparent and regenerative architectures. In this setup, the gNB is functionally split into the Distributed Unit (DU) and the Central Unit (CU). The DU is mounted on-board, while the CU remains on the ground, situated after the NTN gateway. The feeder link in this architecture is based on the F1 interface

protocols. This configuration offers a trade-off: it allows for some level of on-board processing (handled by the DU), while leaving more complex operations to the ground-based CU.

Each of these architectures has its own implications for the implementation of 5G services via NTNs. The choice of architecture depends on several factors, including the complexity of the services offered, the required quality of service, cost considerations, and the technical capabilities of the NTEs. For instance, the transparent architecture is simpler and potentially less costly but may offer limited capabilities in terms of signal processing and quality enhancement. On the other hand, the regenerative and split architectures, while more complex and potentially more expensive, offer greater flexibility and performance benefits.

2.5 Adapting 5G New Radio (NR) to non-terrestrial networks

3GPP's NTN standardization efforts aim to establish global compatibility and a unified user equipment (UE) for seamless connectivity across both terrestrial and non-terrestrial networks (NTN). NTN can operate in two frequency bands: FR1 (<6 GHz) and FR2 (>6 GHz), as shown in FIGURE 9. In FR1, the service link operates at 2 GHz with a maximum bandwidth of 30 MHz, while FR2 utilizes 20 GHz downlink and 30 GHz uplink with a maximum bandwidth of 1 GHz [Vook18].

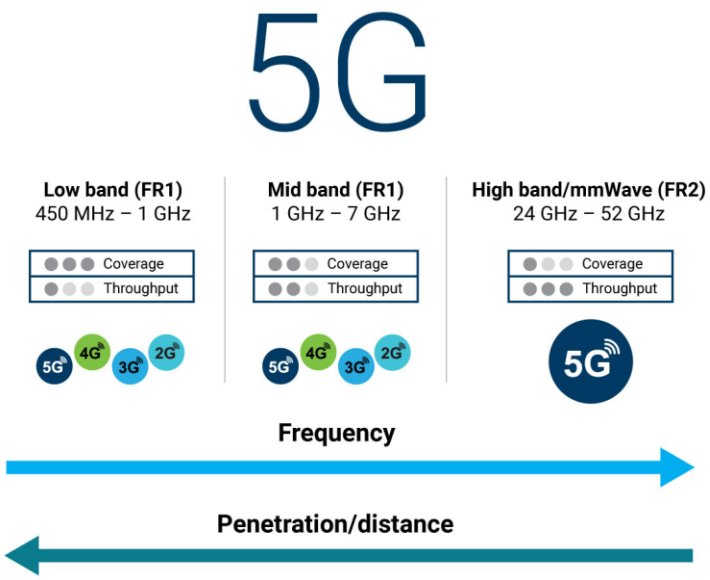

FIGURE 9 5G NEW RADIO FREQUENCY BANDS

Due to the high delays and Doppler shifts associated with NTN, several modifications are proposed to the New Radio (NR) standard. One crucial aspect is UE mobility management, which is affected by the movement of network termination equipment (NTE) relative to the ground. Fixed tracking areas (TAs) can be implemented by steering NTE antenna beams, resulting in simpler feeder link settings when NTEs pass overhead. Alternatively, moving TAs can be employed. Additionally, NR layer 1

modifications include optimized timers, adaptation of the hybrid automatic repeat request (HARQ) procedure, and a switching procedure for the feeder link. Layer 2 modifications involve adaptation of the random access (RA) procedure, timing advance for synchronization, discontinuous reception (DRX), scheduling request (SR), status report extension in radio link control (RLC), sequence numbers in packet data convergence protocol (PDCP), specific system information block (SIB) for NTN (SIB-19), and additional information for UEs in the case of moving cells (i.e., moving TAs), such as satellite ephemeris information.

3. Efficient algorithms for fault tolerant buffers

As discussed in deliverable E5_6G-INTEGRATION3_Innovations, one of the main elements of a switch are the buffers that store packets that are pending processing or transmission [Sei08]. A large amount of the switch memory is devoted to buffering (see for example packet buffers at ucsc.edu) which are critical for performance. This means that buffers account for a significant fraction of the switch area and thus are likely to suffer from errors when operated in a NTN. The standard protection scheme for memories is the use of Error Correction Codes (ECCs) [Chen84]. However, ECCs introduce significant overheads in terms of both additional memory bits, delay, and power consumption. Therefore, it is interesting to find more efficient protection schemes for buffers. In the following different strategies to reduce the cost of buffer protection are presented.

3.1 Reusing packet CRC for buffer protection

A key observation is that the data stored in the buffers is transient in the sense that a packet will be either forwarded to a next hop or drop in a short time, in the scale at most of milliseconds. Therefore, errors in the buffers will not impact the switch functionality in a permanent way. However, this does not mean that errors in the buffers are not important. For example, an error that changes a data bit on a packet that is then transmitted to the destination can have serious effects if the data is critical. In fact, most communication technologies are designed to avoid undetected errors, so that data may or may not be received but a packet that is delivered to the destination contains valid data. For example, in Ethernet standards (see for example Mean time to false packet acceptance (MTTFPA) (ieee802.org)) the probability that a packet with invalid bits is accepted as a valid is designed to be practically zero. This is achieved by implementing error detection mechanisms on the packets such as Cyclic Redundancy Checks (CRC) [Koopman02]. The CRC is computed when the packet is transmitted and checked when it is received. Therefore, a strategy to provide efficient protection to the contents of the buffers is to store the CRC and when computing it for transmission compare it with the value stored. This only adds the storage of the CRC with no additional computation. The CRC has typically 32 bits per packet which is much less than a parity bit per memory. For example, for a 64-bit word memory, the parity introduces an overhead of 1.56% while a 32 bits CRC on an average size packet of approximately 6000 bits is only 0.53%.

3.2 Data compression for buffers

Another potential strategy to reduce the size of buffers is to compress the data they store. However, data compression makes the contents more vulnerable to errors, therefore there is a need for efficient yet error tolerant data compression. To this end, one possibility is to use Tunstall codes and exploit some of their features for error protection, this can be done by modifying the assignment of data patterns to symbols and then adding an additional table in the decompression process [Liu23]. This is achieved by using a conversion table that corrects most of the errors as shown in FIGURE 11.

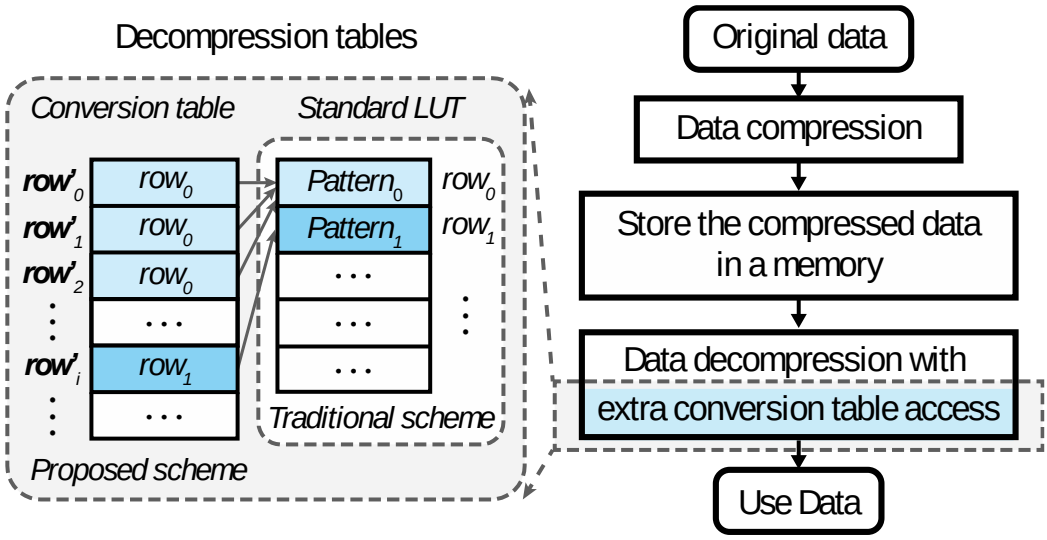

FIGURE 10 ERROR CORRECTION AS PART OF DECOMPRESSION OF TUNSTALL CODES [LIU23]

This enables an end-to-end buffer system that is error protected while implementing compression as shown in FIGURE 11.

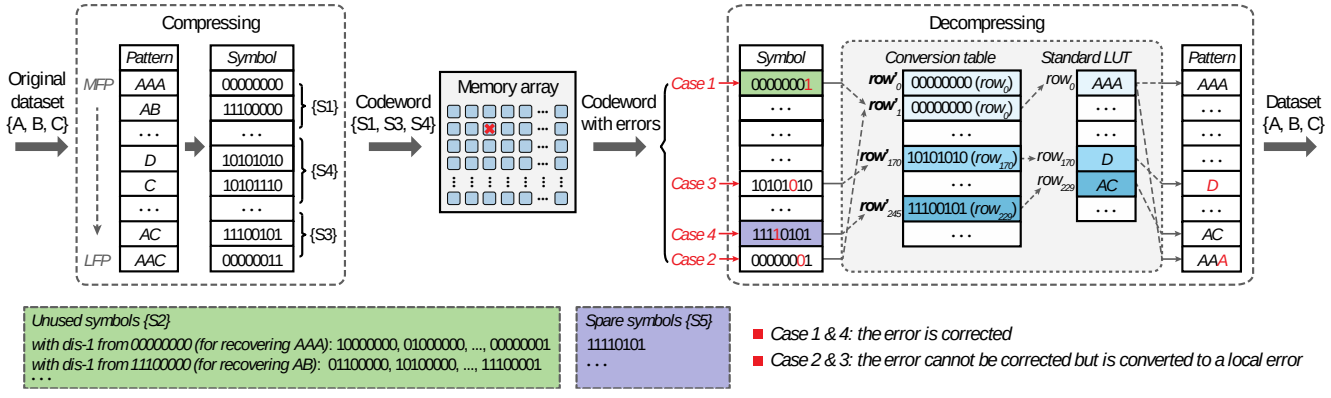

FIGURE 11 OVERALL TUNSTALL CODE-BASE ERROR TOLERANT COMPRESSION AND DECOMPRESSION

3.3 Error tolerant buffer analytics

The analysis of the traffic that traverses the switch is a critical aspect of modern switches and it is drawing an increasing amount of resources [Misa21]. Data sketches are intensively used to support

telemetry functions at scale with a reasonable cost [Yang23] and again a problem is that they are vulnerable to errors. One possibility is to use the properties of the data sketches to detect and correct errors. For example, for sketches that compute the quantiles of the data such as the KLL the ordering of the data can be exploited to detect errors and mitigate them effectively as discussed in [Gao24]. The KLL sketch is illustrated in FIGURE 12, it has a number of layers that compress the original data to extract que quantiles.

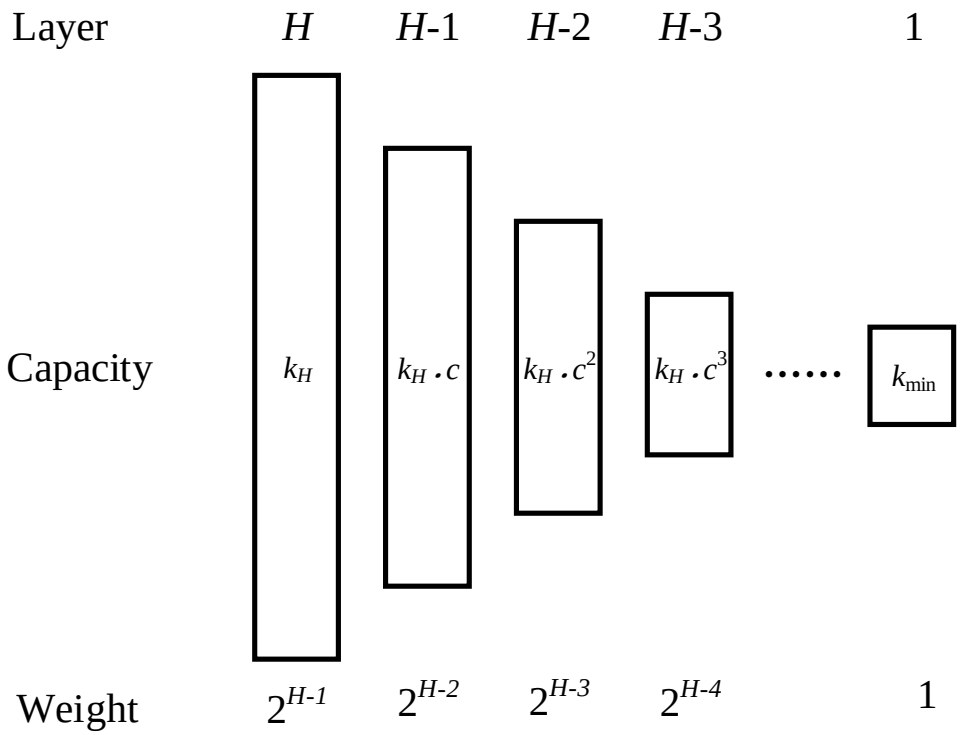

FIGURE 12 STRUCTURE OF THE KLL SKETCH [GAO24]

The data in all layers (except the first layer) of the KLL sketch are ordered as shown in FIGURE 13. Therefore, errors can be detected when this incremental property is not met in a layer. In that case we can replace the faulty data with adjacent data. This protection should be performed for both the construction and query phases. In more detail, the faulty data can be replaced with a linear interpolation of the adjacent data points. This would prevent most errors from affecting the quantiles estimates without requiring the use of additional memory.

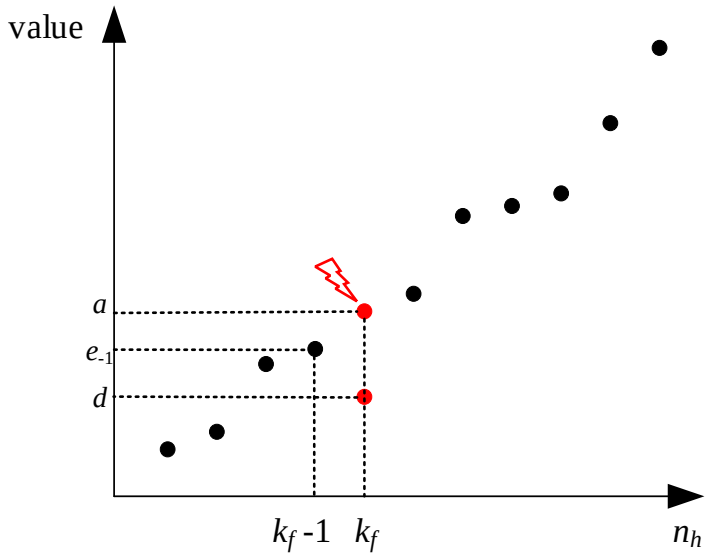

FIGURE 13 DATA STORED IN THE KLL SHOWING HOW THE INCREMENTAL PROPERTY CAN BE USED TO DETECT ERRORS

Similar strategies can be used to protect other data sketches used also in telemetry against errors, for example those used for cardinality estimation such as Hyperloglog (HLL) [Chabhoub10]. In the case of HLL, errors that make a counter value become much smaller than the rest are the ones that have a critical impact on the cardinality estimate. Therefore, protection can be achieved by identifying counters with an abnormally low value and removing them from the estimation as proposed in [Reviriego22]. This enables protection of the HLL sketch with not additional memory bits. Similarly, for the K minimum values sketch that is also used to estimate cardinality, protection can be implemented using the implicit ordering of the K values stored with no additional memory cost. This illustrates how many data sketches can be efficiently protected by reusing their properties. This is an interesting line of research as data sketches are increasingly used in switches and other network devices.

A different approach is to combine different functionality in the same data sketch so that the memory usage is reduce so that ECCs can be added with no additional cost over the original design. This has been done for example by combining approximate membership checking and cardinality estimation in [Reviriego24]. An adaptive cuckoo filter is used for approximate membership checking and at the same time for cardinality estimation achieving good results as shown in FIGURE 14.

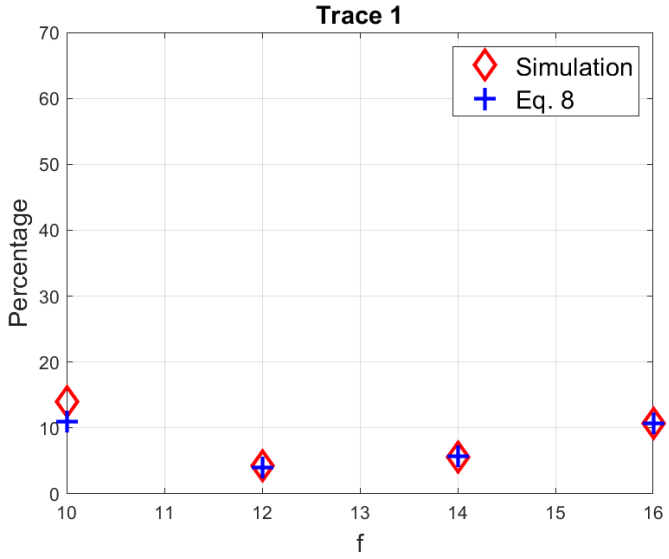

FIGURE 14 CARDINALITY ESTIMATION ERROR FOR A CAIDA TRACE WHEN USING THE APPROACH IN [REVIRIEGO24] AS A FUNCTION OF THE NUMBER OF FINGERPRINT BITS.

3.4 Fault tolerant channelizers

A structure that is closely related to the buffers and it is widely used in onboard communication systems are the channelizers. A transparent digital bent pipe payload is commonly used in multi-beam satellite communication systems [Kawamoto20]. The structure of the channelized is shown in FIGURE 15. 6, in which *N* is the number of beams. For the uplink, the RF signal of each beam is converted to a baseband wideband signal by the down converter (DC) and analog-to-digital converter (ADC). Then parallel digital channelizers extract the narrowband sub-channels in each beam, which are then switched by a router to different downlink beams. The narrowband sub-channels for the same downlink beam are combined into a wideband signal by the synthesizer, which is then converted to a RF signal by digital-to-analog converter (DAC) and up converter (UC). Therefore, *N* channelizers and *N* synthesizers run in parallel for a digital bent pipe payload with *N* beams, so they account for most of the computing resources of the transparent payload.

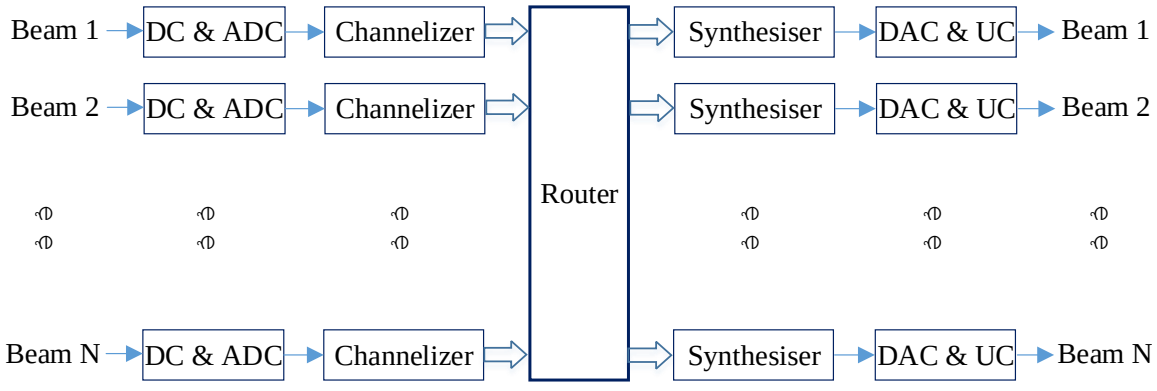

FIGURE 15 STRUCTURE OF A TRANSPARENT DIGITAL BENT-PIPE PLATFORM [GAO23]

The relationships between the signals of the different channels can be used to propose fault tolerant implementations. In more detail, additional redundant channels that are a linear combination of existing channels can be added to the system and used for error detection and correction [Gao23]. The redundant channels are obtained by combining the inputs of the original modules with real weights as

$$\begin{cases} x_{r1} = \sum_{i=1}^{N} a_i x_i \\ x_{r2} = \sum_{i=1}^{N} b_i x_i \end{cases}. \quad (1)$$

This relationship is maintained to the multiple outputs as

$$\begin{cases} z_{r1} = \sum_{i=1}^{N} a_i z_i \\ z_{r2} = \sum_{i=1}^{N} b_i z_i \end{cases}, \quad (2)$$

if all modules run correctly, due to the linear property of the modules, in which z_i and z_r are the outputs of the original modules and the redundant modules, respectively. Then if one of the original modules (e.g. the n-th one) is corrupted by SEU so that its output is changed to $z_n' = z_n + \delta$, then we will have:

$$\begin{cases} \Delta_1 = z_{r1} - \sum_{i=1}^{N} a_i z_i = a_n \delta \\ \Delta_2 = z_{r2} - \sum_{i=1}^{N} b_i z_i = b_n \delta \end{cases}, \quad (3)$$

and the faulty module can be identified by $\Delta_2/\Delta_1 = b_n/a_n$. To ensure the fault detection (for which we need distinct b_n/a_n) and reduce the complexity, we usually make $a_i = 1$ and $b_i = i$ (i = 1, 2, …, N). Finally, the output of the faulty module can be recovered as:

$$z_n^{corrected} = z_{r1} - \sum_{i \neq n} z_i. \quad (4)$$

The overall strategy is illustrated in FIGURE 16, the protection scheme has a lower cost when the number of channels (N) in the system is larger as most of the redundant elements are independent of N. This has been validated in an FPGA implementation using fault injection.

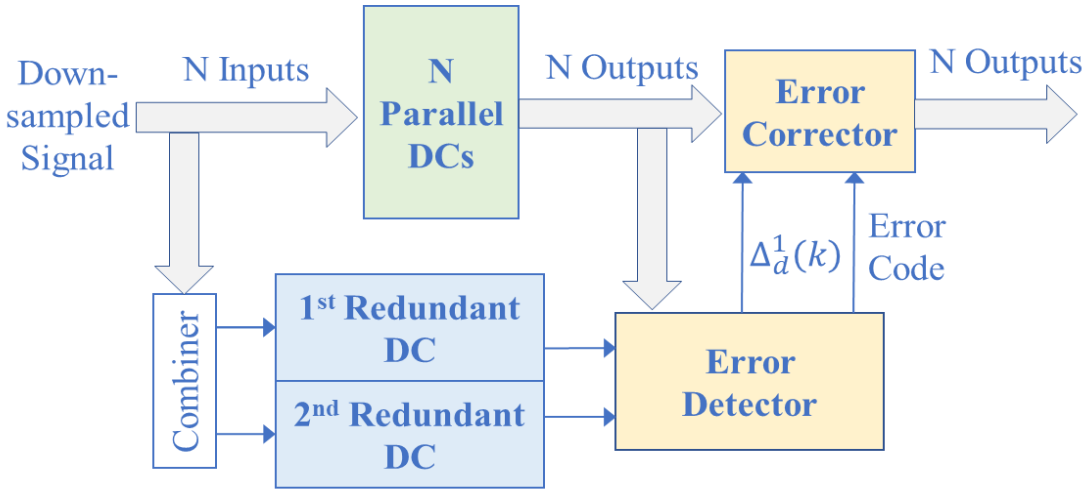

FIGURE 16 FAULT TOLERANT DIGITAL CHANNELIZERS (DGS) [GAO23]

3.5 Fault tolerant channel decoders

Although the channel decoders are not part of the buffers, they play an important role in any satellite-based communication system and thus their fault tolerance is also important. Low Density Parity Check (LDPC) codes [Richardson01] are widely used in modern communication systems and in particular in satellite systems [Zhang21]. The structure of a typical LDPC decoder is shown in FIGURE 17. The decoding is done in a series of message passing steps among the bits of the block until the values converge to the final values that are then checked with parity checks. The computational effort required by this decoding process is large and thus many implementations rely on hardware platforms such as field programmable gate arrays (FPGAs) [Cong22] Unfortunately, SRAM based FPGAs are quite sensitive to errors as they can affect their configuration memory changing the circuit implemented [Gao20].

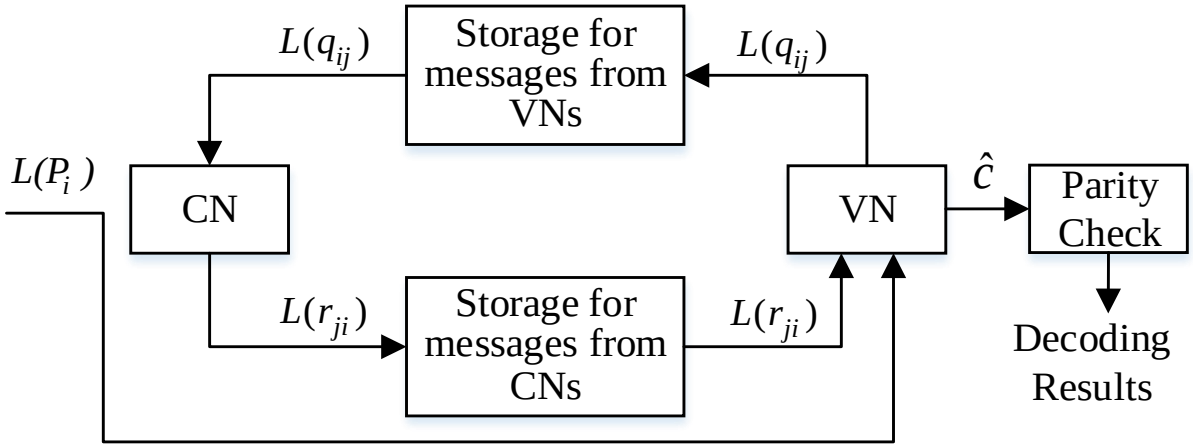

FIGURE 17 STRUCTURE OF AN LDPC DECODER

In [Gao23b] a protection scheme for FPGA implemented LDPC decoder was proposed. It is based on the use of two copies of the decoder running in parallel as illustrated in FIGURE 18. The outputs of the two decoders are compared and used to detect and correct errors. Correction is possible by using the properties of the decoders (for an arbitrary system duplication only can provide error detection) thus significantly reducing the cost of the protection.

The scheme has been implemented on Xilinx FPGAs and tested using a fault injection platform as shown in FIGURE 19. The results show how most errors are detected and correct so providing a protection level similar to triple modular redundancy protection that implies the use of three decoders with a cost that is approximately 50% larger.

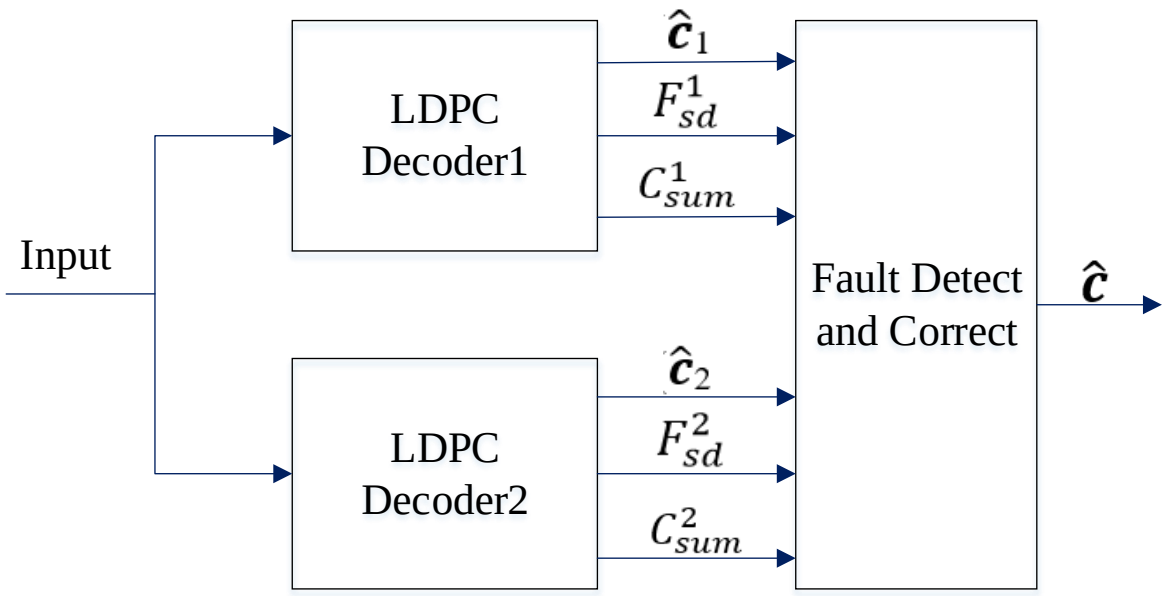

FIGURE 18 FAULT TOLERANT LDPC DECODERS FOR FPGA IMPLEMENTATIONS [GAO23B].

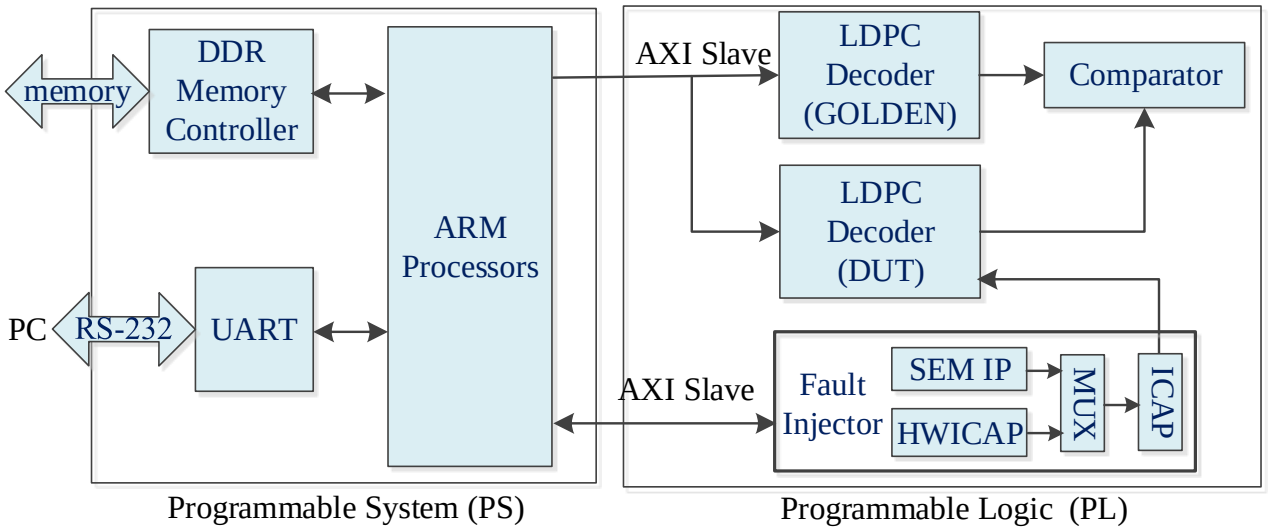

FIGURE 19 FAULT INJECTION ON THE PROPOSED LDPC DECODERS FOR FPGA IMPLEMENTATIONS [GAO23B]

Another type of codes that are widely used in communications is Polar codes [Arikan09] that are also considered for non-terrestrial communications [Fominykh23]. Polar codes rely on a tree structure for message passing that selects the final values of the decoded bits using successive cancellation steps as shown in FIGURE 20. This enables a low complexity decoder implementation. The basic scheme was extended by concatenating a polar code with a cyclic redundancy check (CRC) so that the decoder can more easily identify the correct results. This kind of polar decoder is denoted as CRC-aided SCL (CA-SCL) and is the most widely used polar code decoder at present.

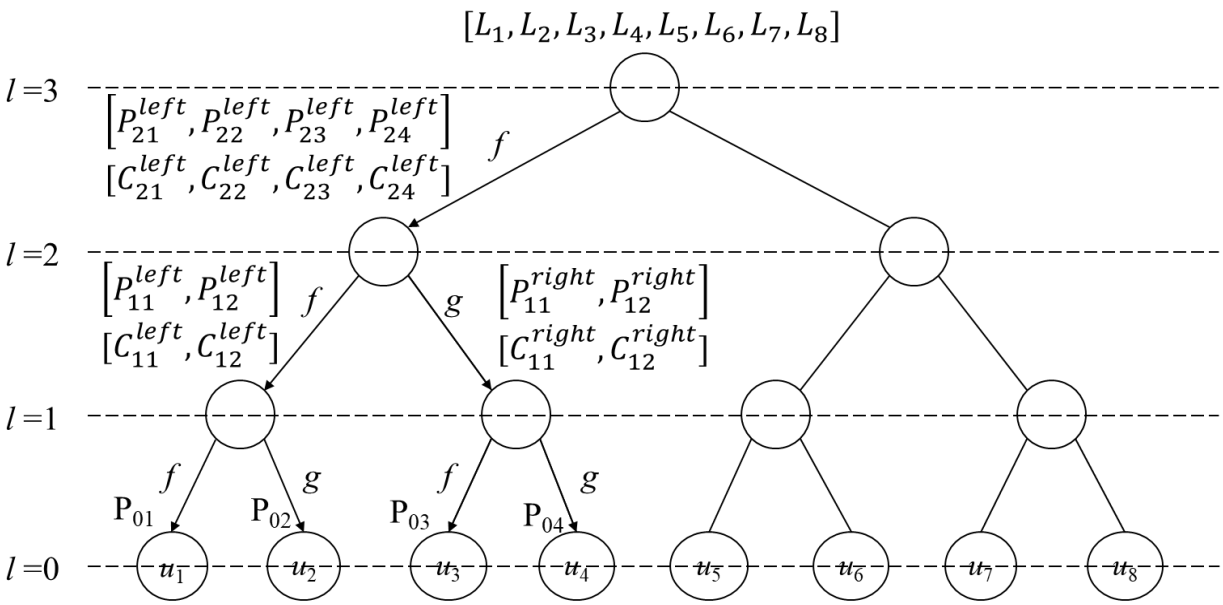

FIGURE 20 TREE STRUCTURE OF A POLAR DECODER

The protection of Polar decoders was considered in [Tian23]. The different parts of the decoder were protected with duplication and re-computation achieving effective protection at low cost. The effectiveness of the protection is illustrated in Table 2 for different block sizes. It can be seen how the vulnerability factor is reduced by several orders of magnitude in all cases.

Table 2. Vulnerability factors of the Polar decoders

	N = 64	*N* = 128	*N* = 256
Protected	5.88E-05	7.54E-06	2.45E-06
Unprotected	2.26E-02	1.45E-02	1.11E-02

4. Traffic analysis and forecasts with ML techniques and impact of caching

The efficient management of network traffic is paramount to ensure seamless communication and data transfer. To achieve this, we harness the power of advanced technologies such as Machine Learning (ML) and Artificial Intelligence techniques and strategic caching mechanisms. In this section, we delve into the fascinating realm of traffic analysis and forecasting, exploring how ML techniques enable us to gain insights into network behavior and predict future traffic patterns. Additionally, we investigate the impact of caching strategies on optimizing network performance, making real-time data retrieval faster and more efficient. These innovative approaches are essential in our ongoing

mission to provide robust and reliable telecommunications services in an ever-evolving digital landscape.

4.1 An overview of ML and AI for Non-Terrestrial Networks

Essentially, Machine Learning (ML) differs from conventional engineering approaches by building domain knowledge from examples, leading to the production of models representing the specific object of study. ML models can adapt to new scenarios, perform well in exceptional cases, and spot trends and patterns. However, ML requires a significant amount of training data, and its accuracy depends heavily on the quality of the training set (FIGURE 21). Processing large data volumes can be time and energy-consuming, posing challenges for adoption in edge and battery-powered devices [Wang09].

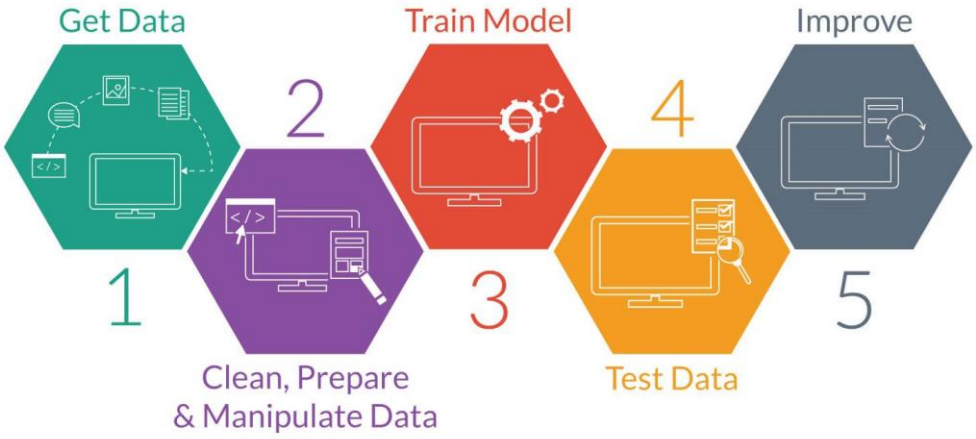

FIGURE 21 MACHINE LEARNING PROCEDURE

There are three main types of ML algorithms (FIGURE 22):

1. Supervised Learning: Involves learning the relation between input and output spaces. Techniques include regression, classification, neural networks, support vector machines, decision trees, and deep learning methods. Applications in NTNs include remote sensing, decoding, demodulation, and network traffic control.
2. Unsupervised Learning: Works with unlabeled inputs and aims to find a set of properties summarizing the data set. Techniques like K-means clustering, Gaussian mixture models, and principal component analysis are used for dimensionality reduction, security, and multicast transmissions.
3. Reinforcement Learning (RL): Involves an agent interacting with the environment and learning from rewards. Techniques include Q-learning, deep Q-networks, and state action reward state action (SARSA). RL is useful in NTNs for optimizing resource allocation, employing non-orthogonal multiple access methods, and dynamic power allocation.

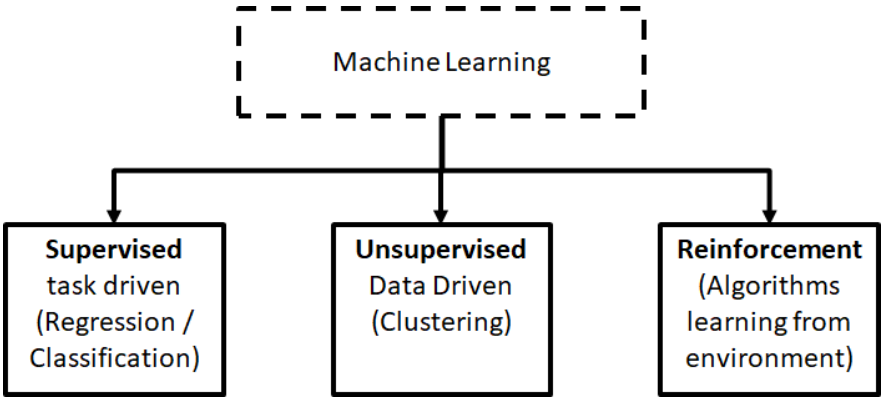

FIGURE 22 TYPES OF MACHINE LEARNING ALGORITHMS

In all cases, the most critical aspect to build accurate ML models is data, both in terms of quantity and quality data. In NTNs, data may come from various network layers (physical, MAC/link, network, and application) and can be used to train ML models. Data includes baseband signals, channel state information, frame error rates, throughput, random access load, latencies, UE battery levels, mobility patterns, traffic load, and user behavior patterns.

4.2 Algorithms for caching most popular contents on HAPS and satellites

There exist multiple algorithms in the literature for caching contents in edge-computing and mobile edge computing architectures [Borst10]. The most popular of them are:

- Least Recently Used (LRU): LRU is a simple and widely used algorithm that evicts the least recently used content from the cache. It is easy to implement and efficient to update, making it a good choice for small to medium-sized caches. However, it can be suboptimal for large caches, as it may evict popular content that has not been recently accessed.
- Least Frequently Used (LFU): LFU evicts the content that has been least frequently accessed. This algorithm is more proactive than LRU, as it attempts to identify and evict content that is not likely to be accessed again in the future. However, it can be more complex to implement and update than LRU.
- Adaptive Replacement Cache (ARC): ARC combines the strengths of LRU and LFU by learning the access patterns of users and adapting its eviction strategy accordingly. This algorithm is more complex than LRU or LFU, but it can significantly improve cache hit rates for large caches.
- Coded Caching: Coded caching is a more advanced approach that leverages network coding techniques to improve cache hit rates and reduce latency. Instead of storing entire files at edge nodes, coded caching stores coded fragments that can be combined to retrieve the original file. This approach can significantly improve the cache hit rate, especially for popular content.
- Machine Learning Based Caching: Machine learning (ML) has been used to develop more intelligent caching algorithms that can dynamically adapt to changing user behavior and network conditions. ML-based algorithms can learn from historical data and current requests

to predict which content is most likely to be requested in the future. This information can be used to select the most appropriate content to cache at each edge node.

The choice of the most appropriate caching algorithm depends on the specific application and network conditions. For small to medium-sized caches, LRU or LFU may be sufficient. For larger caches or applications with dynamic user behavior, more advanced algorithms such as ARC or coded caching may be more beneficial. Machine learning-based algorithms offer the potential for even greater improvements, but they may require more data and computational resources to train and deploy.

In general, considering the limitations of non-terrestrial networks, such as high latency and limited bandwidth, algorithms that prioritize reducing latency and optimizing bandwidth utilization are often considered more suitable. In such a case, the most appropriate caching algorithm for small satellites and HAPS mainly depend on the critical factors and resources of these elements, namely bandwidth, space onboard and energy consumption of the caches and small datacenters installed onboard. It is well-known that massive deployments of small data centers with caching and computing resources in terrestrial central offices comprises massive energy consumption.

4.3 Other applications of AI/ML in Non-Terrestrial Networks

Like many other fields, NTN is expected to be a major advancement in the realm of AI applications. More precise and pragmatic analytical models with reduced overhead consumption, and efficient algorithms with a lower computational complexity are the primary catalysts for the deployment of AI-enabled NTN in next-generation wireless networks. In the following, we give a concise overview of NTN and AI to introduce these two crucial aspects.

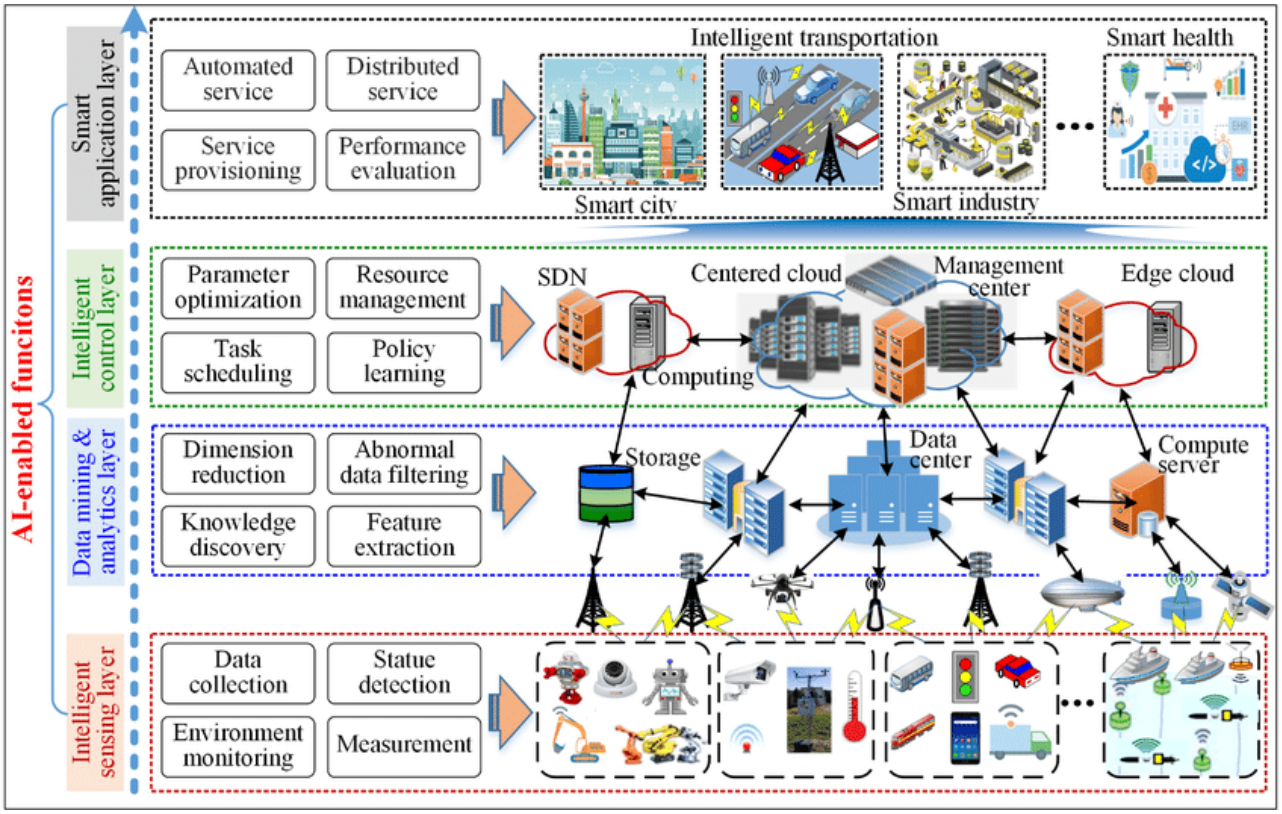

FIGURE 23 AI ENABLED FUNCTIONS IN NON-TERRESTRIAL NETWORKS FOR 5G SUPPORT [YANG20]

NTNs are expected to benefit from AI/ML algorithms, like future wireless networks and other networking related fields where AI/ML have arrived to improve, and even optimize its operations. The key benefits of this integration include more accurate and practical analytical models with less overhead, as well as more efficient algorithms that have lower computational complexity [Yang20]. These improvements are seen as essential for deploying AI within next-generation wireless networks (FIGURE 23).

- Complex Task Automation: In Non-Terrestrial Networks (NTNs), the complexity of communication tasks like resource allocation and satellite management is significantly increased, making manual execution challenging and often unfeasible. The complexity and need for precision in these tasks, especially in satellite network performance optimization, exceed the capabilities of manual operations. However, Machine Learning (ML) and Deep Learning (DL) offer solutions for automating these complex tasks. By using feedback from the network, ML and DL enable not only precise actions but also the automation of complex procedural chains without human intervention.
- Tractable solutions: The deployment of next-generation Non-Terrestrial Networks (NTNs) presents unique challenges due to their complex architecture, notably when integrating satellite networks which adds numerous parameters essential for optimal performance. This complexity can lead to solutions that are either computationally intractable or, if computationally feasible, highly inefficient in terms of resource management within these NTN networks.

Data-driven decision making: Data-driven decision-making in Non-Terrestrial Networks (NTNs) shows promise over traditional probabilistic and deterministic models, which often rely on strong assumptions for general closed-form expressions, leading to significant performance deviations in real-world applications. Machine Learning (ML) models, derived from data, consider various scenarios during training without needing these assumptions, offering a more accurate reflection of real networks. This is particularly valuable in NTNs where additional path-loss components like atmospheric attenuation and scintillation complicate modeling. AI approaches, thus, offer a more precise capture of real scenarios compared to theoretical models.

- Adaptability and learning: AI algorithms enhance the adaptability and learning capabilities of Non-Terrestrial Networks (NTNs) by adjusting to changing network conditions and learning from past experiences. Utilizing Machine Learning (ML) techniques, including Reinforcement Learning (RL) and predictive modeling, AI can progressively optimize network operations and respond to user demands. This enables NTNs to efficiently allocate resources, fine-tune network settings, and proactively address faults. The autonomous decision-making ability of AI allows NTNs to dynamically manage network challenges, boost operational efficiency, and provide consistent service, thus optimizing resource use and ensuring dependable communication in complex and changing environments.
- Reduced computation complexity: Data-driven AI techniques, particularly Deep Learning (DL), offer solutions to the issue of high computational complexity in Non-Terrestrial Networks (NTNs). The challenge lies in the fact that deriving optimal algorithms for NTN's complex systems is difficult, and when achieved, they often require computational resources beyond practical limits due to a multitude of network variables. DL approaches help by simplifying high-dimensional data through feature learning, efficiently extracting implicit features from the network's complexities. This ability to streamline data and reduce computational load makes AI methods especially useful for overcoming the various challenges faced in NTN environments.
- Reduced transmission overhead: AI, particularly through Deep Learning (DL) techniques, offers a way to minimize transmission overhead in Non-Terrestrial Networks (NTNs), which can be burdensome with traditional methods due to extensive information exchange among network entities like satellites and users. For instance, calculating the Doppler shift traditionally requires broadcasting satellite ephemeris data to user equipment (UEs), leading to substantial communication overhead and reduced data rates. DL techniques can estimate the Doppler shift without the need for such information, thereby drastically cutting down on the transmission overhead and enhancing the overall network throughput.
- Real-time implementation: AI techniques, especially online Deep Reinforcement Learning (DRL), are identified as viable for enabling real-time network optimization and management in Non-Terrestrial Networks (NTNs), where decisions must be made within milliseconds to tens of milliseconds. Complex algorithms are impractical for such rapid decision-making and often default to heuristic or offline methods. The use of online DRL allows for adaptable, real-

time decision-making processes that can manage resources efficiently and are crucial for latency-sensitive operations like scheduling and handover decisions in NTNs.

- Leveraging Channel State Information: In communication networks, Channel State Information (CSI) is used by the Base Station (BS) to enhance network performance through the selection of appropriate modulation, channel coding, and other schemes, based on feedback from User Equipment (UE). Machine Learning (ML) methods, such as Reinforcement Learning (RL), can utilize this existing feedback data to train models that optimize these selections. This demonstrates AI's ability to integrate into existing communication infrastructures seamlessly, leveraging CSI without requiring changes to the data transmitted from UE to BS, and without incurring additional overhead costs.

4.4 AI/ML-based optimizations in Non-Terrestrial Networks

The following overviews methodologies for optimizing different aspects of NTNs [Iqbal23]:

A) Beam hopping:

Modern communication satellites use multiple beams to serve a large number of users over extensive areas. This is done through spatial multiplexing into different Non-Terrestrial Network (NTN) cells. Satellites can reuse allocated spectrum with minimal interference and provide strong signals to ground terminals using low power, thanks to beamforming techniques. Traditional satellite communications often use simple fixed beam allocation policies due to high costs and limited onboard processing resources. These methods lack the flexibility to adjust to changing traffic demands.

Beam hopping is a flexible technique for beam allocation, adapting to the varying demands of NTN cells. It involves activating different beams based on current demands, allowing for efficient response to changes in traffic. This process can classify cells into high, medium, and low demand categories, adjusting beam intensity as demands fluctuate.

The challenge of beam hopping involves determining which beams to activate, when, and for how long, to maximize network performance within capacity constraints. This is formulated as an optimization problem, considering metrics like throughput, delay, and fairness, along with power and spectrum constraints. Real networks, however, may find these convex optimization solutions unrealistic and not directly applicable.

For non-convex problems, finding globally optimal solutions is difficult. Sub-optimal solutions are sought using algorithms like steepest gradient descent and heuristic iterative approaches. Meta-heuristic methods like Genetic Algorithm (GA), Simulated Annealing (SA), and Particle Swarm Optimization (PSO) are used for more feasible solutions with less computational complexity.

A significant challenge in beam hopping is the large search space for optimal solutions, especially with satellites having hundreds to thousands of beams. Hence, low-complexity suboptimal solutions are sought, with Deep Learning (DL) approaches emerging as a suitable alternative.

Supervised Learning (SL) has been applied, using labeled datasets of beam hopping patterns with inputs like channel matrix, transmission power, and traffic demand. This involves reducing complex problems to simpler ones for training DL models. Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL) approaches have also been used to minimize transmission delay and optimize throughput and delay fairness. These involve designing state spaces, action sets, and reward functions to train models for dynamic allocation in real-time scenarios.

The AI approaches for beam hopping in NTN encompass a range of strategies, from conventional optimization to advanced AI techniques, addressing the complex challenge of dynamic and efficient resource allocation in satellite networks.

B) Handover optimization

Maintaining an orbital path around Earth, Low Earth Orbit (LEO) satellites move at higher velocities (around 7.8 km/s) compared to Geostationary Orbit (GEO) satellites, orbiting Earth typically within 2 hours. This short visibility time to ground User Equipment (UE) creates challenges for integrating LEO satellites into terrestrial networks, necessitating frequent handovers for continuous data sessions, which in turn, degrades network performance due to communication overhead.

LEO satellites, with smaller coverage areas than GEO satellites, require a large number to achieve global coverage. In dense constellations like Starlink, each UE is covered by multiple satellites, allowing selection of the best one based on various criteria. However, handover decisions in LEO networks are more complex due to limited satellite visibility. Traditional network attachment, based on signal power and quality, must also consider service time for LEO satellites.

A typical handover scenario involves multiple satellites and a single UE. The decision is complex even with just a few satellites, as it must consider network load, channel conditions, and service time. Various strategies are employed to address this, including greedy approaches focusing on maximum service time, signal quality, or minimum network load, but these do not provide optimal solutions.

The handover scenario can be modeled as a directed graph with weights set by criteria like Quality of Service (QoS) and service time. Approaches like bipartite graph matching, network flow-based cost minimization, potential games, and heuristic algorithms have been proposed. Dynamic optimization based on forecasting and channel reservation are also used to design efficient handover algorithms.

Reinforcement Learning (RL) is a natural fit for this problem, with handover criteria as states and UEs as agents making decisions based on network performance. Various approaches, including Deep Reinforcement Learning (DRL) and Multi-Agent Reinforcement Learning (MARL), have been explored. These models consider multiple objectives like satellite load and signal quality. Advanced Deep Learning architectures like Auction based DL, Double Deep Q-Network (DDQN), and Successive DQN are also applied for optimal handover decisions.

However, current systems place handover decision-making at the satellite (BS) level, not the UE, conflicting with existing standards. Distributed multi-agent learning architectures also raise stability concerns in real-world implementations. Future research must efficiently address these challenges,

including careful investigation of handover criteria to provide agents with necessary information for learning environmental mobility behavior.

C) Doppler shift estimation

Low Earth Orbit (LEO) satellites experience significant Doppler effects due to their high-speed movement relative to ground user transceivers. This Doppler shift, caused by the change in frequency of a signal as the satellite and receiver move towards or away from each other, is especially pronounced in LEO systems due to their rapid orbital movement.

The formula for calculating the Doppler shift is $\delta f = f0 \times vc \times \cos(\theta)$, where δf is the Doppler shift, $f0$ is the original frequency, v is the relative velocity, and θ is the angle between the direction of the transceiver and the signal's direction. In LEO satellites, the frequency offset can be significant, for example, 48 kHz at a center frequency of 2GHz. This shift leads to issues like Inter-Carrier Interference (ICI) as user equipment (UE) tunes to different carrier frequencies than initially assigned.

Efforts to characterize the Doppler effect in LEO satellite systems have been ongoing since the advent of communication satellites. Approaches vary from simple equations for satellites in circular equatorial orbits to more complex models considering various orbital parameters. Global Navigation Satellite System (GNSS) can provide satellite positioning data to estimate Doppler shifts for upcoming transmissions, but this adds cost and complexity and is subject to signal weaknesses and interferences.

Recent methods for estimating Doppler shifts in LEO systems include stochastic geometry, Maximum A Posteriori (MAP) estimation, algebraic solutions, and two-stage estimators combining spectral analysis and filtering. Orthogonal Frequency Division Multiplexing (OFDM) carriers in 5G integrated systems also use reference signals to estimate Doppler shifts.

However, these methods are often cumbersome, simplifying satellite orbital mechanics for feasibility, which can impact accuracy. The high-speed, constant movement of LEO satellites leads to time-variant wireless environments, increasing computational complexity for traditional estimation approaches. UEs may require ephemeris data from satellites, adding communication overhead.

Machine Learning (ML) approaches are emerging as practical alternatives for Doppler effect characterization in this context. Channel State Information (CSI) contains information about Doppler shifts due to the mobility of transceivers. ML models, including Fully Connected Neural Networks (FCNN), Multi-Layer Perceptrons (MLP), and hybrid Convolutional Neural Network-Long Short Term Memory (CNN-LSTM) models, have been used to estimate Doppler shifts using various channel characteristics and signal parameters. These techniques are still in early research stages for Non-Terrestrial Networks (NTN).

While Deep Learning (DL) methods show promise in estimating Doppler shifts using channel parameters, analyzing the predictable satellite trajectories could also be a viable approach. A complexity analysis is needed to determine the feasibility of replacing traditional methods with these emerging DL architectures in real-world systems.

D) Spectrum sharing

In the emerging 6G integrated Terrestrial-Non-Terrestrial Networks (TNTN), satellite and terrestrial cellular networks are expected to share the same frequency bands (S and Ka-Band), unlike traditional communication systems where they occupy different bands. This shared spectrum utilization aims to enhance spectral efficiency and improve Quality of Experience (QoE) for users. However, it introduces Co-Channel Interference (CCI) between satellite and terrestrial signals, necessitating innovative spectrum-sharing strategies to minimize interference.

The TNTN environment presents a hierarchical network of non-terrestrial and terrestrial Base Stations (BSs), requiring efficient spectrum-sharing methods. Traditional approaches like frequency reuse, directional antennas, and adaptive power control can mitigate CCI, but they either consume more spectrum or increase system complexity.

To address these challenges, spectrum sensing in cognitive radio networks has been introduced. This allows unlicensed users to detect the occupancy status of a target band using methods like Energy Detection (ED), Cyclo-Stationary Detection (CSD), and Eigen Value-based Detection (EVD). However, these methods have limitations; ED is simple but performs poorly in low Signal-to-Noise Ratio (SNR) scenarios, while CSD and EVD are effective but computationally complex.

Machine Learning (ML) is increasingly adopted for spectrum sharing, offering reduced computational complexity and capturing correlation in integrated satellite-terrestrial networks. Various intelligent learning approaches have been developed for TNTN networks:

- Convolutional Neural Network-Bidirectional Long Short-Term Memory (CNN-BiLSTM) Model: Used for developing spectrum-sharing strategies by analyzing historical occupancy data of Geostationary Orbit (GEO) satellite spectrum for Low Earth Orbit (LEO) satellite users.
- CNN-LSTM-Based Spectrum Sensing Method: Helps satellites capture spatial and temporal correlations in spectrum occupancy effectively.
- CNN-LSTM for Frequency Assignment Prediction: Predicts frequency assignments for satellites based on historical data.
- Modified Q-Learning in Reinforcement Learning (RL): Adapts access and modulation schemes for Non-Geostationary Orbit (NGSO) satellites in NGSO-GEO systems.
- Support Vector Machine (SVM) and CNN-Based Spectrum Prediction: Combines low-complexity spectrum sensing with CNN-based prediction models using historical data.
- Cooperative Multi-Agent Deep Reinforcement Learning (MADRL): Manages bandwidth in a game-theoretic model to minimize inter-beam interference.
- CNN-Based Spectrum Reconstruction: Reconstructs spectrum from incomplete data for satellite networks.

Despite these advancements, real-time spectrum sensing decisions are crucial for enhancing the throughput of secondary users. This necessitates the replacement of conventional Long Short-Term Memory (LSTM) architectures with more efficient, low-complexity Echo State Network (ESN) architectures. Additionally, future research should focus on spatial spectrum sharing, complementing the current temporal sharing approaches, to fully leverage the potential of 3D Space-Air-Ground Integrated Networks (SAGIN).

E) Resource allocation and sharing

Power and spectrum are essential resources in wireless networks, including Non-Terrestrial Networks (NTN). Spectrum allocation typically involves assigning carriers of equal width from the allocated spectrum, optimizing their number and positions for good signal quality using minimal resources. This process, often achieved through orthogonal splitting (frequency reuse), aims to maximize spectral efficiency. However, perfect orthogonality is not always attainable, leading to Co-Channel Interference (CCI), which can be mitigated by increasing transmission power. Yet, due to power constraints and the need for energy efficiency, indefinite power increase is not viable.

Radio resource management in NTNs thus faces the challenge of balancing power and spectrum resources. This task often involves complex, non-linear, and non-convex optimization problems, particularly challenging due to factors like Signal to Interference and Noise Ratio (SINR) and mixed-integer programming aspects of carrier assignment. Traditional convex optimization methods fall short due to high computational complexity.

To address these challenges, suboptimal and metaheuristic approaches are employed, tackling parts of the problem separately and iteratively tuning parameters. Various suboptimal methods optimize resource allocations in satellite systems, while heuristic and metaheuristic approaches like Genetic Algorithms (GA) and Particle Swarm Optimization (PSO) aim for suboptimal solutions more efficiently.

Machine Learning (ML), particularly Deep Learning (DL) and Reinforcement Learning (RL), is increasingly adopted to address resource allocation in satellite networks. ML approaches complement conventional optimization algorithms, reducing computational complexity by narrowing down feature spaces. For instance:

- Deep Learning Frameworks: Combined with conventional algorithms, these frameworks reduce computation complexity and feature space, aiding in resource allocation.
- Model-free Deep Reinforcement Learning (DRL): Utilized for power allocation in high-throughput satellites, offering more efficient solutions.
- Q-learning-based Long-term Capacity Allocation: Implemented in RL frameworks for heterogeneous satellite networks, focusing on capacity allocation over time.
- Actor-Critic and Critic Only RL Frameworks: Applied for optimal resource allocation in LEO satellite networks, balancing multiple factors efficiently.
- Advanced RL Frameworks: Including DRL, Multi-objective DRL, and Multi-Agent DRL, these frameworks are proposed for tackling resource allocation challenges in NTNs.

As power and spectrum are both critical and limited resources in NTNs, ongoing research emphasizes the need for new DL architectures to jointly allocate these resources efficiently. The integration of AI approaches in resource allocation for TNTN is pivotal, offering potential solutions for the complex trade-offs between spectrum efficiency, power consumption, and overall network performance.

F) Computation offloading

Satellite-terrestrial integrated networks are increasingly vital for enhancing the computational capabilities of existing terrestrial network architectures, particularly for applications like Augmented Reality (AR) and Virtual Reality (VR) that require high data processing and extremely low latency. Traditionally, terrestrial base stations (BSs) are sparsely deployed due to high costs, leading to challenges in meeting the high data processing demands of these applications. To address this, computation tasks are often offloaded to terrestrial clouds via satellites. However, this can introduce significant latency, especially with Geostationary Earth Orbit (GEO) satellites due to their longer propagation delay.

The emergence of Low Earth Orbit (LEO) satellites, which have comparatively lower propagation delays, offers a solution. These satellites can act not just as relays but also as edge servers, processing tasks themselves. This leads to a three-level hierarchical architecture comprising ground User Equipments (UEs) connected to terrestrial BSs, LEO satellites, and terrestrial clouds. In this setup, terrestrial BSs can offload computational tasks to both LEO satellites and terrestrial clouds via LEO satellites, potentially reducing latency.

The main challenge in this task offloading process is balancing the need to meet low-latency requirements of applications while minimizing energy consumption of the satellites. This scenario is typically formulated as an optimization problem, seeking efficient offloading approaches for integrated TNTN architecture. Various conventional methods like 3D hypergraph matching, game theory, stochastic approaches, and efficient algorithms have been proposed to solve this problem. Joint optimization frameworks that consider both task offloading and resource allocation have also been explored.

However, these traditional algorithms have limitations, particularly in real network scenarios. Some approaches fail to consider cooperation between terrestrial clouds and LEO satellite servers, leading to suboptimal solutions. Additionally, these methods can be highly dependent on network states, resulting in significant overhead, and often require many iterations to converge, leading to high computational complexity.

To address these challenges, Machine Learning (ML) approaches, especially Deep Learning (DL) and Reinforcement Learning (RL), are being explored for task offloading problems. Deep Reinforcement Learning (DRL)-based frameworks that depend on channel state information, dynamic queue conditions in satellites, and decentralized solutions using Deep Q-Network (DQN) and Double DQN (DDQN) have been proposed. Deep Deterministic Policy Gradient (DDPG) algorithms are used to address optimization problems in DQN frameworks, including potential security issues. Long Short-Term Memory (LSTM) models are utilized to consider channel conditions and energy dynamics, and DL-based caching strategies are explored in satellite edge networks.
Multi-agent architectures, both in distributed and cooperative environments, are being considered to improve overall system performance. These architectures are used to generate discrete offloading decisions in a supervised manner. As delay constraints vary with network traffic types, offloading decisions need to be made considering these variations. Future research could focus on how computational offloading can be optimized for various network traffic types to enhance network

performance, potentially offering significant improvements in latency and efficiency for satellite-terrestrial integrated networks.

G) Routing

Satellite-terrestrial integrated networks are increasingly vital for enhancing the computational capabilities of existing terrestrial network architectures, particularly for applications like Augmented Reality (AR) and Virtual Reality (VR) that require high data processing and extremely low latency. Traditionally, terrestrial base stations (BSs) are sparsely deployed due to high costs, leading to challenges in meeting the high data processing demands of these applications. To address this, computation tasks are often offloaded to terrestrial clouds via satellites. However, this can introduce significant latency, especially with Geostationary Earth Orbit (GEO) satellites due to their longer propagation delay.

The emergence of Low Earth Orbit (LEO) satellites, which have comparatively lower propagation delays, offers a solution. These satellites can act not just as relays but also as edge servers, processing tasks themselves. This leads to a three-level hierarchical architecture comprising ground User Equipments (UEs) connected to terrestrial BSs, LEO satellites, and terrestrial clouds. In this setup, terrestrial BSs can offload computational tasks to both LEO satellites and terrestrial clouds via LEO satellites, potentially reducing latency.

The main challenge in this task offloading process is balancing the need to meet low-latency requirements of applications while minimizing energy consumption of the satellites. This scenario is typically formulated as an optimization problem, seeking efficient offloading approaches for integrated TNTN architecture. Various conventional methods like 3D hypergraph matching, game theory, stochastic approaches, and efficient algorithms have been proposed to solve this problem. Joint optimization frameworks that consider both task offloading and resource allocation have also been explored.

However, these traditional algorithms have limitations, particularly in real network scenarios. Some approaches fail to consider cooperation between terrestrial clouds and LEO satellite servers, leading to suboptimal solutions. Additionally, these methods can be highly dependent on network states, resulting in significant overhead, and often require many iterations to converge, leading to high computational complexity.

To address these challenges, Machine Learning (ML) approaches, especially Deep Learning (DL) and Reinforcement Learning (RL), are being explored for task offloading problems. Deep Reinforcement Learning (DRL)-based frameworks that depend on channel state information, dynamic queue conditions in satellites, and decentralized solutions using Deep Q-Network (DQN) and Double DQN (DDQN) have been proposed. Deep Deterministic Policy Gradient (DDPG) algorithms are used to address optimization problems in DQN frameworks, including potential security issues. Long Short-Term Memory (LSTM) models are utilized to consider channel conditions and energy dynamics, and DL-based caching strategies are explored in satellite edge networks.

Multi-agent architectures, both in distributed and cooperative environments, are being considered to improve overall system performance. These architectures are used to generate discrete offloading decisions in a supervised manner. As delay constraints vary with network traffic types, offloading decisions need to be made considering these variations. Future research could focus on how computational offloading can be optimized for various network traffic types to enhance network performance, potentially offering significant improvements in latency and efficiency for satellite-terrestrial integrated networks.

H) Slicing

Network slicing is a process in wireless networks where the physical network is virtually partitioned into different "slices," each tailored to specific service requirements. These slices are dynamically allocated radio resources based on user demand within each slice. This approach allows multiple network services to share the same physical infrastructure while ensuring that each service receives the necessary resources to maintain a minimum service level. Network slicing is particularly advantageous in networks with varied traffic patterns, such as integrated Terrestrial-Non-Terrestrial Networks (TNTN), where it can significantly enhance the performance and efficiency of the network.

In TNTN networks, network slicing can address the diverse needs of different use cases, like massive Machine Type Communications (mMTC) and enhanced Mobile BroadBand (eMBB) applications. A typical scenario in such networks might involve a satellite and a terrestrial base station forming multiple slices to cater to various user groups. For example, one slice could be dedicated to high-priority users, utilizing resources from both the satellite and terrestrial base station. Another slice might focus on users with low latency requirements, primarily served by the terrestrial base station, while a third slice could cater to remote users accessible only via the satellite.

The goal in network slicing is to optimize a composite utility function that considers various network performance characteristics (like average throughput) and costs (like slice reconfiguration and resource reservation costs). This function forms the objective in an optimization problem, with constraints typically being the minimum service level requirements for different types of services across slices. While simple heuristic approaches have been tested for real-time implementation, they often fall short of guaranteeing optimal performance.

To improve performance and ensure optimal resource allocation, Artificial Intelligence (AI)-based approaches have been proposed. For instance, one approach considers a Radio Access Network (RAN) slicing problem where the objective is to balance bandwidth and spectrum consumption while satisfying Quality of Service (QoS) and inter-slice isolation constraints. Deep Learning (DL) architectures have been tested in simple 2-slice satellite-terrestrial integrated network scenarios to address these challenges.
Another approach adopts Machine Learning (ML) methodologies similar to meta-heuristic Ant Colony Optimization (ACO) to realize network slicing in TNTN environments. Furthermore, Deep Reinforcement Learning (DRL) frameworks have been applied to air-ground integrated networks, using algorithms like Deep Deterministic Policy Gradient (DDPG). In these frameworks, both actor and critic networks are typically Fully Connected Neural Networks (FCNNs) with multiple layers.

Looking to the future, distributed learning architectures are considered promising for real network implementations. These architectures could enable more efficient and effective network slicing in complex and dynamic TNTN environments, tailoring network resources to specific service requirements while ensuring optimal utilization of the available infrastructure. Network slicing, especially with AI and ML innovations, is poised to play a crucial role in the evolution of TNTN networks, enabling them to meet diverse and demanding service requirements more effectively.

I) Channel estimation

Channel estimation in Non-Terrestrial Networks (NTNs) is crucial for network planning and interference management. It involves estimating the channel's impact on transmitted signals in a wireless environment, typically represented by Channel State Information (CSI). Traditional methods like Minimum Mean Square Error (MMSE) or Least Squares (LS) are computationally intensive and not always practical for real-time network operations. The challenge in NTNs is exacerbated by long propagation delays and rapidly changing environments, making timely CSI acquisition difficult.

Consequently, Machine Learning (ML)-based methods are gaining traction as a more effective alternative for channel estimation. These methods can treat channel estimation as a Supervised Learning (SL) problem, using features like distance, time delay, received power, azimuth Angle of Arrival (AoA) and Departure (AoD), elevation angle, Root Mean Square (RMS) Delay Spread, and frequency as inputs, with CSI as the output labels.

For example, it can be studied the reciprocity property of downlink and uplink channels in Time Division Duplexing (TDD) systems to estimate the downlink channel from uplink CSI using a Long Short-Term Memory (LSTM)-based Deep Learning model. Another approach could be to estimate CSI from historical data using a Convolutional Neural Network (CNN)-LSTM model. However, channel estimation being a near real-time process necessitates exploration of low-complexity Neural Networks, such as Echo State Networks (ESNs), for practical and realistic implementations in NTNs.

5. Analysis of 3GPP EdgeApp and ETSI MEC to deal with the NTN segment

Multi-access Edge computing (MEC) is where core network and cloud computing capabilities are moved to the "edge" of the network closer to the customers, reducing the physical distance for communications [Filali20].

MEC offers application developers and content providers cloud-computing capabilities and an IT service environment at the edge of the network. This environment is characterized by ultra-low latency and high bandwidth as well as real-time access to radio network information that can be leveraged by applications. MEC provides a new ecosystem and value chain. Operators can open their Radio Access Network (RAN) edge to authorized third-parties (FIGURE 24), allowing them to flexibly and rapidly deploy innovative applications and services towards mobile subscribers, enterprises and

vertical segments. MEC in NTNs also involves considerations for deployment positions, energy constraints, processing capacity, storage issues, and network element compatibility.

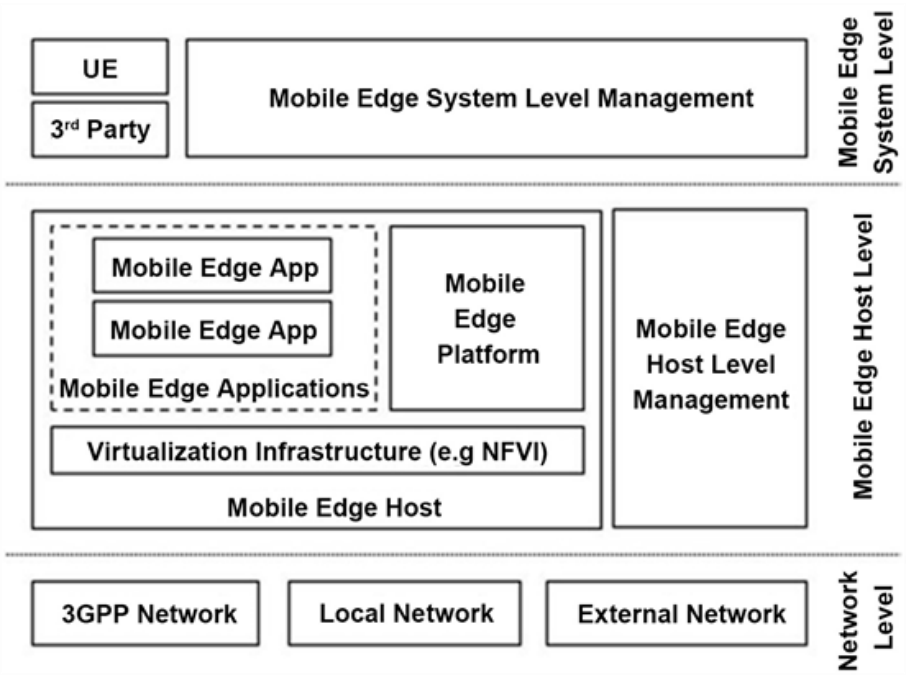

FIGURE 24 ETSI MEC ARCHITECTURE

5.1 Strategic relevance of MEC

MEC is a natural development in the evolution of mobile base stations and the convergence of IT and telecommunications networking. Multi-access Edge Computing will enable new vertical business segments and services for consumers and enterprise customers. Some of its use cases include:

- V2X
- video analytics
- location services
- Internet-of-Things (IoT)
- augmented reality
- optimized local content distribution and
- data caching

It uniquely allows software applications to tap into local content and real-time information about local-access network conditions. By deploying various services and caching content at the network edge, Mobile core networks are alleviated of further congestion and can efficiently serve local purposes. It is worth noting that MEC also addresses fixed and WLAN accesses.

MEC industry standards and deployment of MEC platforms will act as enablers for new revenue streams to operators, vendors and third-parties. Differentiation will be enabled through the unique applications deployed in the Edge Cloud [Makkena23].

MEC currently focuses on its 'Phase 3' activities that consider a complex heterogeneous cloud ecosystem. This work embraces MEC security enhancements, expanded traditional cloud and NFV Life Cycle Management (LCM) approaches, and mobile or intermittently connected components and consumer-owned cloud resources.

5.2 From MEC to EdgeAPP

3GPP EDGEAPP is a framework developed by the Third Generation Partnership Project (3GPP) to enable the deployment and operation of edge applications in 5G networks. EDGEAPP is defined by the 3GPP for enabling Edge Applications 3GPP TS 23.558. Edge computing has been a major focus area in 3GPP since Rel-17, with the following working groups:

- SA6: Application layer architecture, and deployment scenarios (FS_EDGEAPP, EDGEAPP)

- SA2: System Architecture enhancement for supporting Edge Computing (enh_EC, eEDGE_5GC)

- SA3: Security aspects for supporting SA2 eEDGE_5GC and SA6 EDGEAPP (FS_eEDGE_Se)

- SA5: Management aspects on Edge Computing (FS_Eedge_Mgt)

EDGEAPP provides a standardized way for application developers to deploy their applications on edge nodes, which are located closer to the user equipment (UE) to reduce latency and improve the user experience.

The EDGEAPP framework (FIGURE 25) consists of three layers:

- Edge Enabler Layer (EEL): This layer provides the basic capabilities for managing and deploying edge applications, including the ability to discover, connect to, and switch between edge nodes.
- Edge Application Server (EAS): This layer hosts the application logic and data. It can be deployed on a variety of edge nodes, such as base stations, local data centers, or even on the UE itself.
- Application Client (AC): This layer is the software running on the UE that connects to the EAS and consumes the application's services.

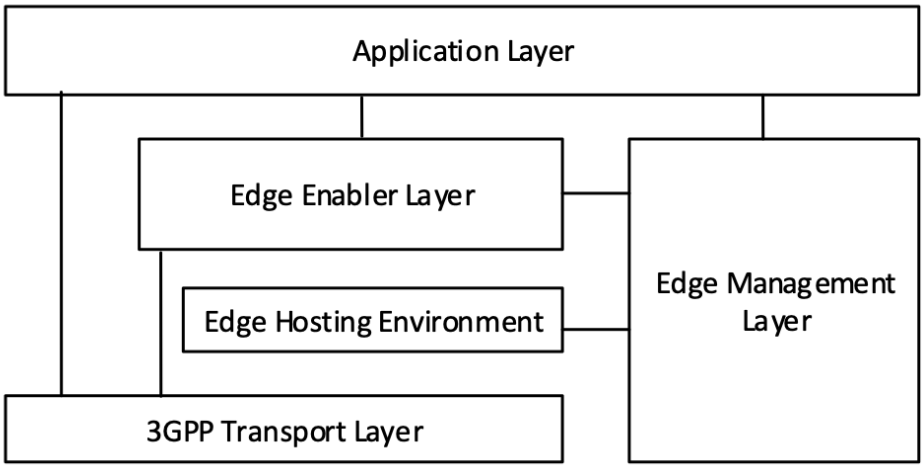

FIGURE 25 EDGEAPP FRAMEWORK

The EDGEAPP framework (FIGURE 25) enables a number of benefits for edge applications, including:

- Reduced latency: Applications hosted on edge nodes can provide a much lower latency to the UE compared to applications hosted in a central cloud.
- Improved performance: Applications can take advantage of the proximity to the UE to improve their performance, such as by reducing network congestion and improving bandwidth utilization.
- Enhanced security: Applications can be deployed on more secure networks, such as the private mobile network (PMN), which can help to protect sensitive data.

The EDGEAPP framework is an important step forward in the development of edge computing for 5G networks. It provides a standardized way to deploy and manage edge applications, which will help to accelerate the adoption of edge computing for a wide variety of applications, for example:

- Real-time video streaming: Edge applications can be used to stream real-time video content, such as live sports or concerts, to UEs without the need to send the entire stream to a central cloud.
- Low-latency gaming: Edge applications can be used to provide low-latency gaming experiences, reducing the perceived lag between user input and game actions.
- Industrial automation: Edge applications can be used to collect and analyze data from industrial sensors and actuators, enabling real-time monitoring and control of industrial processes.

As 5G networks continue to evolve, the EDGEAPP framework will play an increasingly important role in enabling the deployment of edge applications that can provide a wide range of benefits to users and businesses [Ali22, Vukobratovic22].

5.3 Edge Artificial Intelligence

Edge artificial intelligence (AI), or AI at the edge, takes one step further with respect to edge computing since it is the implementation of artificial intelligence algorithms in an edge computing environment. This allows computations to be done close to where the data is generated, rather than at a centralized cloud computing facility or an offsite data center [Vukobratovic22].

With Edge AI, AI/ML algorithms will be able to be deployed (as microservices for example or EdgeAPP) directly in the HAPS or satellites. Algorithms trained with data directly deployed at the HAPS will allow faster response. A use-case example could be in surveillance, where HAPS or drones equipped with powerful cameras can detect abnormal behaviours (violence, robbery, etc), take shots and even call directly for assistance or police. This is especially important in scenarios and remote areas where fiber connectivity is not possible, like in forests, mountains, seaside, etc.

5.4 ML and AI innovations to be deployed on HAPS and Satellites

To enhance interoperability, compatibility, and innovation in 5G networks, the Third Generation Partnership Project (3GPP) has been developing standards and specifications for AI and edge computing. TS 23.501 outlines the overall architecture of the system and the roles of the various network functions and interfaces that support AI and edge computing capabilities. TS 23.791 examines traffic distribution across multiple access networks, such as cellular, Wi-Fi, and satellite, while TR 23.793 focuses on the enhancements of the NWDAF, which collects, analyzes, and provides network data for AI applications and services. Finally, TS 23.735 studies architecture enhancements and requirements for supporting edge computing in 5G networks, such as edge application enablement, edge discovery, and edge mobility.

Applications of ML and AI algorithms in the NTN segment include, but are not restricted, to the following use case applications [Iqbal23]:

1. Moving Cell Connectivity: RL and Q-learning algorithms optimize the position of high-altitude platform stations (HAPS) and low Earth orbit (LEO) satellites to maximize network capacity and minimize transmission latency.
2. Unserved/Under-served Areas: RL techniques find the best position for NTNs, with a focus on energy resource management and multi-user multi-access edge computing (MEC) offloading.
3. Throughput Increase: ML assists in 3D positioning and content caching/storage, improving latency performance and network capacity.
4. Multi-Connectivity: RL algorithms select the most convenient pair of transmitters to maximize throughput in hybrid NTN/terrestrial networks.
5. Secondary Link Backup: ML classifies network failures and predicts radio link outages, aiding in service restoration and load balancing.
6. Disaster Relief: ML optimizes UAV positions for coverage and power efficiency. Computer vision and deep learning techniques assist in rescue operations.

7. Broadcasting/Multicasting: ML optimizes cache performance, making content delivery more efficient.

6. Evaluation of Round-Trip Times in LEO satellite constellations

One of the potential advantages of Low Earth Orbiting (LEO) satellite constellation networks is the reduction in delay, which is critical for some services and applications. Theoretically, LEO satellite networks have a lower propagation delay when the endpoints are at distances longer than a few thousand km due to the higher propagation speed of radio transmission compared to fiber. However, the theoretical analysis makes assumptions about transmission paths that are not valid in all cases and does not account for other factors that influence delay, such as processing or queuing delays. Therefore, experimental studies based on measurements and simulations are required. We conducted the first comparison between real Round Trip Time (RTT) measurements on terrestrial networks and simulated RTTs on different LEO constellations using the Hypatia [Kassing20] simulator and real fiber measures provided by Wondernetwork. The results confirm the potential benefits of LEO constellations not only in reducing RTT values, but also in reducing their dispersion versus distance.

In recent years, the world has witnessed a massive deployment of LEO constellations to enhance connectivity worldwide, with hundreds of satellites launched by companies like Amazon's Kuiper, Starlink, Telesat, or OneWeb. The development of massive LEO satellite constellations holds the promise of bringing high-speed Internet access everywhere [Ahmmed22], since it is estimated that more than 40% of the world's population does not have regular access to the Internet. These megaconstellations are expected to provide connectivity to both deep rural and remote locations, and complement existing fiber-based terrestrial networks with the so-called non-terrestrial networks (NTN) toward 5G and 6G everywhere.

These mega constellations are designed with hundreds or thousands of satellites, each covering a small part of the planet are envisioned to be provided with Inter-Satellite Links (ISLs) that will create a network in space [Pan19].

The main constellations that are currently in operation are SpaceX Starlink, Amazon Kuiper, Telesat, and OneWeb, each with their own design in terms of the number of satellites, altitude, and number of shells. The altitudes are in the 550-1,325 km range, and the number of shells is between two and four, with each shell having several orbits on which tens of satellites are placed. This configures a large network with thousands of nodes and links that constantly move relative to the terrestrial endpoints, which makes its operation a complex task.

One of the key metrics of network performance is delay, and more specifically RTT. A large RTT affects the TCP throughput, further degrades the quality of interactive services, and can even make them unfeasible [Kurose21]. Traditionally, satellite communications have suffered from large RTT because of the use of Geostationary Earth Orbiting (GEO) satellites. The geostationary orbit is at an altitude

of 36,000 km and therefore the RTT of communication over a GEO satellite is approximately 480 ms. However, with LEO constellations, the situation is completely different, satellites are now at an altitude within 500 to 1,500 Km and the delay of the uplink or downlink is only a few milliseconds. In fact, for distant endpoints, a LEO satellite network theoretically has less RTT than a terrestrial network. This is because radio waves propagate at the speed of light (i.e. 300,000 km/s) while optical signals propagate only at two-thirds of this due to silica fiber refraction index (i.e. 200,000 km/s over silica fibers). Hence, in some distant cases (nation-wide), satellite communications can be faster (in terms of propagation delay) than fiber transmission, even taking into account the extra distance of the LEO satellites' altitude.

The previous discussion has focused on the propagation delay, however, there are other factors that influence the RTT such as the processing or queuing delays that are not taken into account in the theoretical analysis. Even propagation delay can deviate significantly from theory, as the packets will not follow the shortest geographical path either in satellite or in terrestrial networks. Therefore, to better understand the potential benefits of LEO networks in reducing RTTs, a more realistic analysis based on real measurements or simulation models that capture relevant details is needed.

6.1 RTT in fiber optic networks

In terrestrial networks, transmission generally occurs over optical fiber links on which the propagation speed is approximately 200,000 km/s which introduces a propagation delay of 5 µs per km. The propagation delay also depends on the path followed by the packets that may deviate significantly from the shortest path, for example, due to the presence of geographical obstacles such as deserts, mountains, or sea [Martinez23]. This can be clearly seen by looking at the long-distance fiber connections in operation. This leads in some cases to transmission distances much longer than the shortest geographical orthodromic distance and increases the propagation delay. The other delay components, such as queueing or processing delays, depend on the number of hops a packet traverses. Queuing delay is also strongly dependent on the network load, and operators try to ensure that there is sufficient capacity to avoid high loads. This is feasible in terrestrial networks as the transmission speed can be upgraded per link when a given utilization is reached. As for processing delay, the continuous advances in electronic technology and the increasing adoption of optical switching have also reduced the delay over the last decade.

In our analysis of terrestrial RTT, we used real ping measurements between different end-points worldwide provided by the WonderNetwork measurement platform. Measurements are taken among more than 100 monitoring stations distributed around the world and include minimum, average, and maximum RTT values at different times of the day for different days in 2022. We shall take the minimum RTT value as the most accurate approximation for propagation delay only (without variable delay contributors like queueing and processing delay).

6.2 RTT in LEO constellations

In satellite networks, the propagation speed is 300,000 km/s, this is 50% higher than in terrestrial networks. On the other hand, the minimal distance over a satellite network is longer, since packets need to go up to a satellite at the origin and then down from the satellite to the destination [Conde24]. As shown in FIGURE 26, the RTT on the satellite path is shorter than in the terrestrial path when $RTT_{sat} \leq RTT_{terrestrial}$ which, in FIGURE 26 (top) occurs when:

$$\frac{4\sqrt{h^2 + \left(\frac{d}{2}\right)^2}}{c} \leq \frac{2d}{\frac{2}{3}c}$$

which occurs when d ≥ 4*h/√5 ≈ 1.79h. Thus, LEO satellite propagation orbiting at 1,000 km is faster for terrestrial distances larger than 1,790 km.

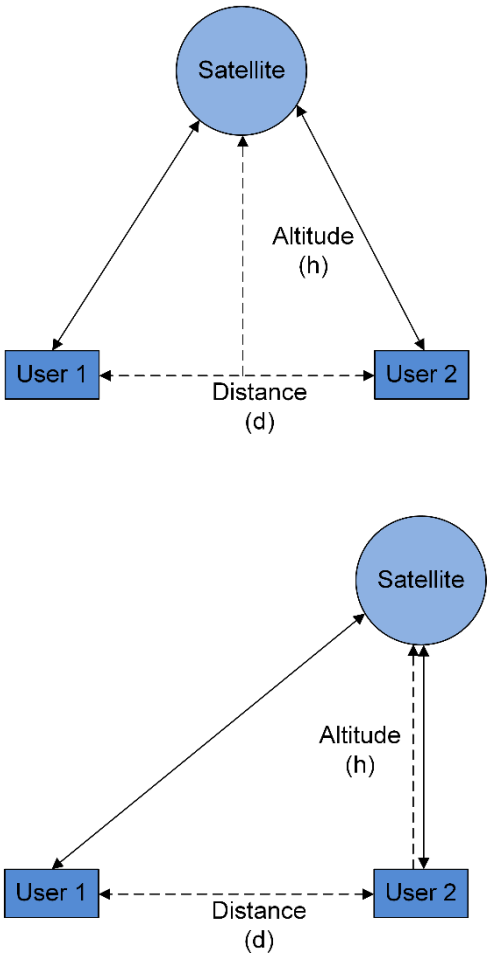

FIGURE 26 ILLUSTRATION OF DIFFERENT SCENARIOS FOR TWO USERS AT A DISTANCE D VIA LEO SATELLITES AT ALTITUDE H

The distance travelled by packets in FIGURE 26 (bottom) is different than that of FIGURE 26 (top). Essentially, the calculation of accurate propagation delay depends on the real-time positioning of satellites in space and stations on Earth.

In the case of LEO networks there are not publicly available RTT measurements that can be used to compare with long and short distances terrestrial RTTs but there are open-source simulators that can provide accurate results, such as Hypatia. With the Hypatia simulator, we computed accurate propagation delays for some orbital shells of three of the major LEO megaconstellations: Starlink, Kuiper, and Telesat, as shown in Table 3. This simulator models the positions of satellites over short periods of time and the paths followed by packets, so the RTT values obtained should capture well the propagation delay between endpoints. The simulation implements ISLs only among neighboring satellites on the same shell. This is reasonable, as implementing ISLs between different shells is challenging due to the different speeds of the satellites. The need for a simulator arises from the absence of real measurements for LEO ISLs communications and the complexity of the scenario, which makes it difficult to conduct a theoretical analysis. The configurations of the constellations are the ones in Table 3. Therefore, they can be used for an initial comparison with terrestrial RTT measurements. The queuing delay with real traffic is not accounted for in the simulations. This has been taken into account when comparing the results, as we explain in the following section.

Table 3. Parameters of the orbital shells studied in the experiment

Constellation	Altitude (km)	Inclination (º)	Satellites
Starlink	550	53.00	1,584
Kuiper	630	51.90	1,156
Telesat	1,015	98.98	351

6.3 Comparison of RTT in LEO networks and fiber networks

For terrestrial networks, the minimum measured RTT value between the stations is used. If measurements are collected over an extended period, as is the case with our study, it is possible to estimate the queueing delay as the difference between the average RTT and the minimum RTT. In other words, the minimum RTT can be approximated as the RTT without queuing delay. This is reasonable as in both cases, the dispersion of the RTT values for two given locations was limited. The analysis of RTT fluctuations is left for future work.

In FIGURE 27, we show the RTTs for many source-destination pairs worldwide to obtain an initial understanding of the differences between terrestrial and LEO networks and also between LEO constellations. The x-axis is the distance between source and destination, and the y-axis represents the RTT for different satellite constellations (Hypatia simulation RTTs) and real Wondernetwork's measured RTT over terrestrial networks. The red and black dashed lines show the lowest possible propagation delay across the shortest distance for fiber and satellite propagation, respectively. It can be seen that LEO constellations have lower RTT values except for very short distances. The figure also illustrates that the dispersion of terrestrial RTT values versus distance is much larger than for LEO constellations.

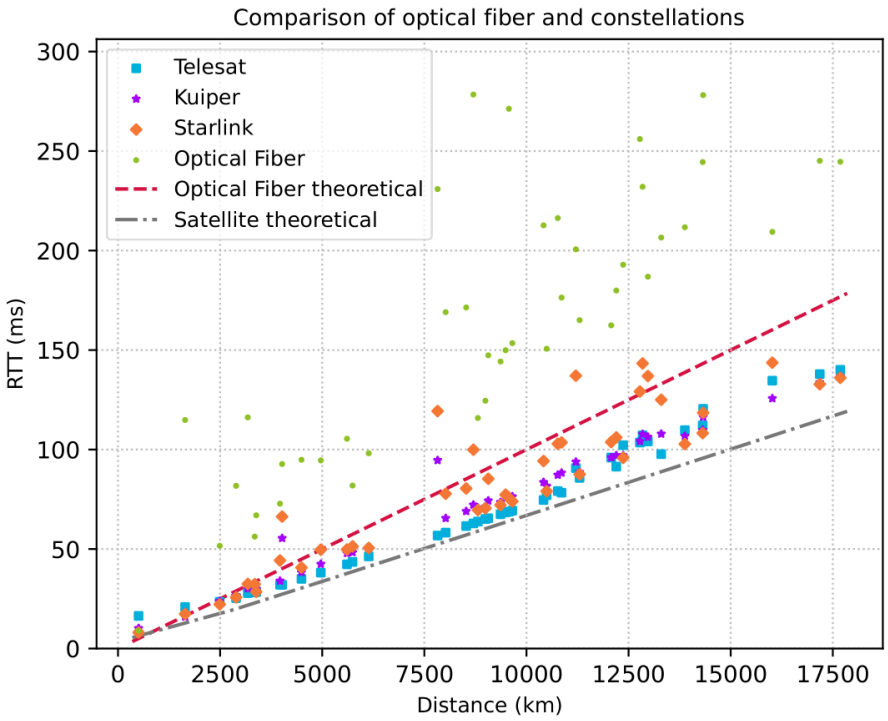

FIGURE 27 RTT VS ORTHODROMIC DISTANCE FOR DIFFERENT CITIES WHEN USING A TERRESTRIAL NETWORK AND THE LEO CONSTELLATIONS [CONDE24]

In a second scenario, we select a few cities and evaluate their RTT to other cities around the world. In more detail, we choose Madrid, New York, Cairo, and Bogota as source cities and Sydney, Barcelona, Ryhad, Los Angeles, Tokyo, and Mexico DF, as destination examples covering multiple continents. The results are shown in FIGURE 28. As shown, LEO constellations have lower RTT values than terrestrial networks, except in short-distance cases like Madrid-Barcelona (600 km apart) and similar pairs of cities. Another interesting observation is that satellite RTTs are closer to their lowest bound than terrestrial RTTs. For example, considering Cairo, the RTT to Ryhad is much larger than to Madrid, which is approximately twice the distance. The same applies when comparing the RTTs between Madrid and Bogotá and Los Angeles. This can be explained since terrestrial networks do not always follow the shortest path (orthodromic distance) due to geographical accidents, geopolitical situation, etc., while in satellites it is easier for radio waves to follow the shortest geographical path.

As an example, there is an orthodromic distance of 3,177 km between Bogota and Mexico City. However, the results reveal that it has a higher fiber-RTT compared to the communication between Bogotá and New York (128.63 ms and 95.96 ms, respectively), even though the orthodromic distance between Bogotá and New York is slightly larger (4,017 km). The traceroute tool reveals that the packets between Mexico DF and Bogotá do not follow a direct route and they traverse Mexico DF, Houston, Miami, and Bogotá, covering approximately 7,073 km (14,144 km round trip). On the other hand, the path between Bogotá and New York goes through a submarine cable from Mexico DF to Ashburn (US) and from Ashburn to New York, resulting in a shorter distance of approximately 5,175

km (10,350 km round trip). A similar effect is observed in geographically isolated locations such as Tokyo or Sydney. For example, in the communication between Tokyo and Cairo (separated by an orthodromic distance of 9,570 km), packets traverse the following route: Tokyo to San Jose (US) via a submarine cable, San Jose to Ashburn across the US, Ashburn to Paris via another submarine cable, Paris to Marseille (France) through Europe, and Marseille to Cairo through another submarine cable. All of this results in the actual distance traveled by the packet being approximately 22,000 km (44,000 km round trip), which is 2.3 times larger than the orthodromic distance.

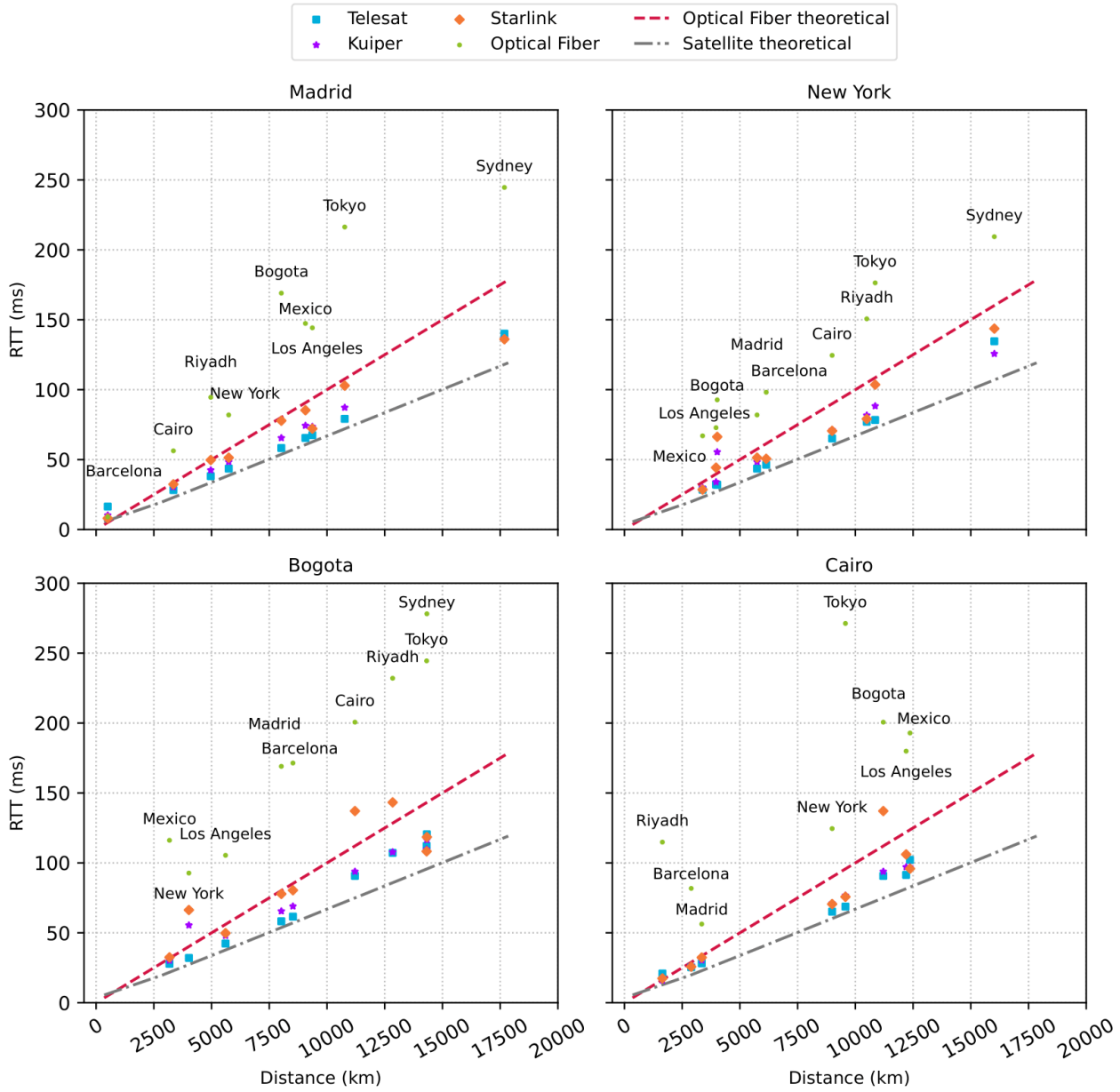

FIGURE 28 RTTS FOR DIFFERENT CITIES OVER LONG DISTANCES WHEN USING A TERRESTRIAL NETWORK AND THE LEO CONSTELLATION [CONDE24]

Finally, we conduct a last experiment to compare cities that are at short distances, smaller than 1,500 km, and thus could have lower RTTs when using terrestrial networks. As in the previous experiment, we picked four cities and, for each of them, we compared the RTTs to a few other cities. The results are summarized in FIGURE 29. It can be seen that for London, the terrestrial network has lower RTT than LEO constellations for short distances. Instead, in the case of Venice, which is also on the same continent and in a developed country, the RTT of the terrestrial network is larger for all destinations (except Rome) because for all inter-country links, it traverses Milan and Marseille causing an extra mean delay of 14.55 ms (SD = 2.4 ms). This shows that LEO constellations may also have an advantage for locations that are not placed in the core of the network, as packets may have to travel across a national network to reach the core, which can add a significant distance. This is even clearer when we look at the results for Limassol in Cyprus, where RTT is almost independent of geographical distance, which is explained because it takes all packets 48.7 ms (SD = 3.4 ms) to leave the island. A similar situation is observed in Cairo, where terrestrial networks again have much larger RTTs for all distances. In the case of Cairo, two distinct phenomena are observed. On the one hand, there are routes that are relatively close geographically, such as Cairo to Athens (with an orthodromic distance of 1,118 km), but packets follow a very long path. In this case, they pass through Marseille via a submarine cable and traverse the southern part of Europe to reach Athens, covering approximately 5,021 km (10,042 km round trip). On the other hand, there are routes that are geographically close but connected through slow networks, as is the case with Jerusalem (orthodromic distance of 425 km), where the distance travelled by the packets is approximately 650 km (1,300 km round trip), but the latency is much higher than the theoretical limit (108.06 ms and 4.25 ms respectively). This situation can probably be generalized to other countries that are not technologically developed where the RTT is higher [Martinez23]. These results suggest that even for short distances, LEO networks may have an advantage due to the limited terrestrial connectivity among many locations.

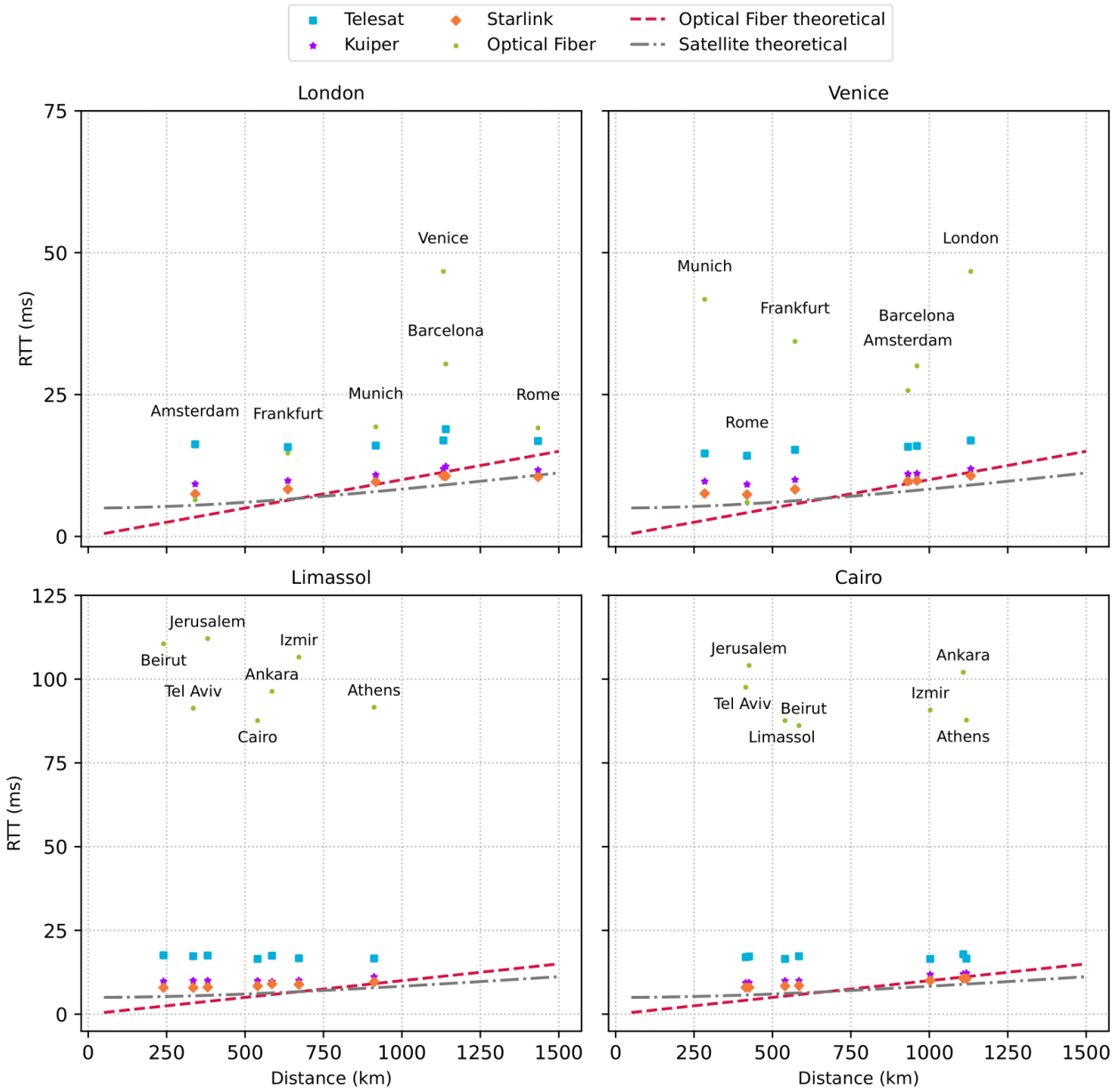

FIGURE 29 RTTS FOR DIFFERENT CITIES OVER SHORT DISTANCES WHEN USING A TERRESTRIAL NETWORK AND THE LEO CONSTELLATIONS [CONDE24]

From the comparison, the potential of LEO networks to reduce RTTs is confirmed. However, the results should be taken with caution as for LEO networks simulation values are used, while for terrestrial networks we have used real measurements that capture additional delay components, including different traffic conditions, variability, processing and transmission delay, etc. The differences in RTT among constellations and the dispersion of values are also interesting results of the comparison. The results confirm the potential of LEO networks to reduce not only the RTT values but also their dependence on the location of the endpoints. LEO constellations reduce RTT over long distances but also over short distances, especially in countries with poor geographical connections or countries with little technological development. The main limitation of our work is to perform a comparison between real measurements (terrestrial) and simulated measurements (satellites). In the

future, when LEO constellations implement ISLs, the experiment can be repeated with real data, considering all types of delays and network congestion levels.

7. Conclusions

This document is focused on different ways to provide innovations in the integration of HAPS and satellites into 3GPP Terrestrial and Non-Terrestrial Networks.

Section 2 reviews the fundamentals of the physical layer in LTE and its evolution into 5G technologies, focusing on the role of Orthogonal Frequency-Division Multiplexing (OFDM) and the various modulation schemes supported in 5G for achieving efficient, reliable, and speedy network communication. LTE radio frames and Physical Resource Block (PRB) are also reviewed along with Hybrid Automatic Repeat reQuest (HARQ) protocol which combines the error detection mechanisms of ARQ with the error correction of Forward Error Correction (FEC) for enhanced data transmission reliability. Cloud Radio Access Network (Cloud RAN) and Coordinated Multi-Point (CoMP) are also overviewed as transformative technologies enabling more dynamic and efficient network operations, two promising paradigms for improving bandwidth performance and user experience, especially in urban environments.

In general, integrating HAPs and satellites into the 5G ecosystem poses unique challenges and network reengineering paradigms, some of them includes dealing with the latency of HAPs and satellites, reliability of the channel and ensuring service continuity and ubiquitous connectivity in underserved areas. The adaptation of 5G services to NTNs involves re-evaluating and modifying these services to cater to the distinct capabilities and constraints of NTNs, thus extending the reach of advanced communication technologies to previously inaccessible environments.

Section 3 focuses on the technical aspects of network switches and buffers to go onboard satellites and HAPs, and how to deal with errors, especially in space due to solar radiation. Traditional Error Correction Codes (ECCs) secure buffer contents but introduce considerable overhead in memory, delay, and power. On the other hand, reusing packet CRC for transmission verification requires storing the CRC value without additional computation, offering a leaner solution compared to ECCs. Another approach is to use data compression, like Tunstall codes, to reduce buffer size while maintaining error tolerance. This method includes a conversion table for error correction during decompression, enabling an efficient and protected end-to-end buffer system. Additionally, error-tolerant buffer analytics utilize properties of data sketches, such as KLL sketches, to detect and correct errors without extra memory overhead. Techniques like linear interpolation or exclusion of abnormal values enhance the reliability of telemetry data.

Also, in onboard communication systems, fault-tolerant channelizers and channel decoders play a vital role. Channelizers, which separate and route signals in multi-beam satellite systems, can employ redundancy for error detection and correction. By adding linearly combined redundant channels, the system can identify and correct faults, validated by FPGA implementations. Similarly, channel decoders, such as LDPC and Polar decoders used in satellite systems, are protected using duplication

and re-computation strategies. These methods provide cost-effective protection, demonstrated by reduced vulnerability factors in FPGA-based implementations. The text emphasizes the importance of robust and efficient error-correction mechanisms to ensure the integrity and performance of modern communication systems.

Section 4 outlines the utilization of Machine Learning (ML) and Artificial Intelligence (AI) in the optimization and management of Non-Terrestrial Networks (NTNs). ML and AI are pivotal in creating models that can adapt to new scenarios and identify trends. However, they require ample and quality training data, and the processing of such data can be resource-intensive. Concerning algorithms, three primary ML algorithms are highlighted: Supervised Learning, which includes methods like neural networks and decision trees for tasks such as decoding and network traffic control; Unsupervised Learning, which finds hidden patterns in data and is used for dimensionality reduction and security; and Reinforcement Learning (RL), which is applied to optimize resource allocation and power distribution. The latter can provide ample applications in networking scenarios, including NTNs. However, all AI/ML-based algorithms depend on the quality and availability of sensored data from various layers of the network, which can be used to train these models.

Finally, this section also discusses specific applications and use-case scenarios of AI/ML in NTNs, underscoring their potential to revolutionize future wireless networks through the automation of complex tasks, offering tractable solutions, and facilitating data-driven decision-making. AI's adaptability is particularly noted in dynamically managing network challenges and optimizing operations. Techniques like online Deep Reinforcement Learning (DRL) are considered for real-time network optimization, which is crucial for latency-sensitive operations. Furthermore, AI methods can reduce computational complexity and transmission overhead, making them highly effective for NTN environments.

Section 5 discusses Multi-access Edge Computing (MEC) which brings core network and cloud computing capabilities to the edge of the network, closer to users, which results in ultra-low latency, high bandwidth, and real-time access to radio network information. This innovative approach enables operators to offer their Radio Access Network (RAN) edge to third parties, fostering a new ecosystem for the rapid deployment of applications and services for mobile subscribers, enterprises, and vertical markets. MEC is integral to various applications such as vehicle-to-everything communication (V2X), video analytics, and augmented reality, facilitating optimized content distribution and data caching.

MEC is crucial for the telecom industry, offering a platform for new business segments and enabling a unique set of applications for consumers and enterprise customers. It is undergoing 'Phase 3' activities that involve enhancing MEC security, expanding cloud and Network Functions Virtualization (NFV) lifecycle management approaches, and integrating mobile components and consumer-owned cloud resources. This phase is paving the way for new revenue opportunities for operators, vendors, and third-parties through differentiated Edge Cloud applications. Moreover, the 3GPP EDGEAPP framework is a standardization effort to facilitate the deployment of edge applications, providing a structured approach to managing edge applications via three layers: the Edge Enabler Layer, the Edge Application Server, and the Application Client. These advancements promise reduced latency,

improved performance, and enhanced security for a range of applications, from real-time video streaming to industrial automation, as 5G networks evolve. Edge AI further extends the capabilities of edge computing, deploying AI/ML algorithms as microservices or through EdgeAPP in High Altitude Platform Stations (HAPS) or satellites for faster, localized data processing and response, critical in areas without fiber connectivity.

Finally, section 6 focuses on latency estimation experiments in Low Earth Orbit (LEO) satellite constellations and its comparison against fiber delays. These constellations offer lower propagation delays over long distances compared to terrestrial networks due to the faster speed of radio transmission through space versus optical signals in fiber. Nonetheless, a purely theoretical approach falls short of capturing the full picture, as it often overlooks factors such as processing and queuing delays. To bridge this gap, empirical studies incorporating simulations, like those conducted using the Hypatia simulator, and real-world data from Wondernetwork are essential. These studies have demonstrated the superiority of LEO networks in reducing RTT and its variability with distance, especially above 1,000 km separation between end points, underscoring the importance of experimental validation over theoretical analysis.

The global push for ubiquitous connectivity is manifest in the deployment of massive LEO satellite constellations by major corporations, aiming to provide high-speed internet access even in the most remote regions. These constellations are designed to form a dynamic network with thousands of satellites connected via Inter-Satellite Links (ISLs), presenting a complex yet promising solution for global coverage. With operational altitudes ranging from 550 to 1,325 km and multiple orbital shells, these constellations could significantly lower RTT compared to traditional geostationary satellites, which suffer from higher RTTs due to their much greater distance from Earth. Real-world measurement and simulation tools like Hypatia are critical for understanding the true impact of these constellations on network performance, including RTT, and can provide a foundation for comparison against terrestrial network delays. The promise of LEO networks extends not only to long-distance communication but also to shorter links, especially in areas with suboptimal terrestrial infrastructure, highlighting their potential to deliver low-latency connectivity across the globe.

References

[Ahmmed22] T. Ahmmed, A. Alidadi, Z. Zhang, A. U. Chaudhry, and H. Yanikomeroglu, “The digital divide in canada and the role of leo satellites in bridging the gap,” IEEE Communications Magazine, vol. 60, no. 6, pp. 24–30, 2022.

[Ali22] A. Ali, S. Rahman Khan, S. Sakib, M. S. Hossain and Y. -D. Lin, "Federated 3GPP Mobile Edge Computing Systems: A Transparent Proxy for Third Party Authentication With Application Mobility Support," in IEEE Access, vol. 10, pp. 35106-35119, 2022, doi: 10.1109/ACCESS.2022.3162851.

[Arikan09] E. Arikan, "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels," in IEEE Transactions on Information Theory, vol. 55, no. 7, pp. 3051-3073, July 2009.

[Borst10] S. Borst, V. Gupta and A. Walid, "Distributed Caching Algorithms for Content Distribution Networks," 2010 Proceedings IEEE INFOCOM, San Diego, CA, USA, 2010, pp. 1-9, doi: 10.1109/INFCOM.2010.5461964.

[Campana23] R. Campana, C. Amatetti and A. Vanelli-Coralli, "O-RAN based Non-Terrestrial Networks: Trends and Challenges," 2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Gothenburg, Sweden, 2023, pp. 264-269, doi: 10.1109/EuCNC/6GSummit58263.2023.10188308.

[Chabchoub10] Y. Chabchoub and G. Heébrail, "Sliding HyperLogLog: Estimating Cardinality in a Data Stream over a Sliding Window," 2010 IEEE International Conference on Data Mining Workshops, Sydney, NSW, Australia, 2010, pp. 1297-130

[Chen84] C. Chen, and M. Y. Hsiao. "Error-correcting codes for semiconductor memory applications: A state-of-the-art review." IBM Journal of Research and development 28.2 (1984): 124-134.

[Conde24] J. Conde, G. Martinez, P. Reviriego and J. A. Hernandez “Round Trip Times (RTTs): Comparing Terrestrial and LEO Satellite Networks”, in IEEE Conference on Innovation in Clouds, Internet and Networks (ICIN) 2024.

[Cong22] Z. Cong and G. Feng, "A High-speed Satellite Internet Physical Layer LDPC Decoder Design and FPGA-based Implementation," 2022 11th International Conference on Communications, Circuits and Systems (ICCCAS), Singapore, Singapore, 2022, pp. 131-135.

[Filali20] A. Filali, A. Abouaomar, S. Cherkaoui, A. Kobbane and M. Guizani, "Multi-Access Edge Computing: A Survey," in IEEE Access, vol. 8, pp. 197017-197046, 2020, doi: 10.1109/ACCESS.2020.3034136.

[Fominykh23] A. A. Fominykh and A. A. Ovchinnikov, "Comparative Analysis of Polar and LDPC Codes in Space and Satellite Communication Systems," 2023 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF), St. Petersburg, Russian Federation, 2023.

[Gao20] Z. Gao et al., "Fault tolerant design of large-scale digital beam forming in SRAM-FPGAs for software defined satellite platforms," in China Communications, vol. 17, no. 7, pp. 224-235, July 2020.

[Gao23] Z. Gao, J. Xiao, Q. Liu, A. Ullah and Pedro Reviriego, "A Methodology for the Design of Fault Tolerant Parallel Digital Channelizers on SRAM-FPGAs", IEEE Transactions on Circuits and Systems I, vol. 70, no. 5, pp. 2003-2015, May 2023.

[Gao23b] Z. Gao, Y. Cheng, Q. Liu, A. Ullah and P. Reviriego, "Efficient Protection of FPGA Implemented LDPC Decoders Against Single Event Upsets (SEUs) on Configuration Memories," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 70, no. 9, pp. 3770-3780, Sept. 2023.

[Gao24] Z. Gao, J. Zhu and P. Reviriego, "Reliability Evaluation and Fault Tolerant Design for KLL Sketches," in IEEE Transactions on Emerging Topics in Computing (in press).

[Geraci23] G. Geraci, D. López-Pérez, M. Benzaghta and S. Chatzinotas, "Integrating Terrestrial and Non-Terrestrial Networks: 3D Opportunities and Challenges," in IEEE Communications Magazine, vol. 61, no. 4, pp. 42-48, April 2023, doi: 10.1109/MCOM.002.2200366.

[Huang16] Jun Huang, Fangming Ruan, Ming Su, Xiaohong Yang, Shunli Yao and Junhua Zhang, "Analysis of orthogonal frequency division multiplexing (OFDM) technology in wireless communication process," 2016 10th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID), Xiamen, 2016, pp. 122-125, doi: 10.1109/ICASID.2016.7873931.

[Iqbal23] A. Iqbal et al., "Empowering Non-Terrestrial Networks With Artificial Intelligence: A Survey," in IEEE Access, vol. 11, pp. 100986-101006, 2023, doi: 10.1109/ACCESS.2023.3314732.

[Irram20] F. Irram, M. Ali, Z. Maqbool, F. Qamar and J. J. Rodrigues, "Coordinated Multi-Point Transmission in 5G and Beyond Heterogeneous Networks," 2020 IEEE 23rd International Multitopic Conference (INMIC), Bahawalpur, Pakistan, 2020, pp. 1-6, doi: 10.1109/INMIC50486.2020.9318091.

[Kassing20] S. Kassing, D. Bhattacherjee, A. B. Aguas, J. E. Saethre, and A. Singla, “Exploring the internet from space with hypatia”, in Proceedings of the ACM Internet Measurement conference, 2020, pp. 214–229.

[Kawamoto20] Y. Kawamoto, T. Kamei, M. Takahashi, N. Kato, A. Miura and M. Toyoshima, “Flexible Resource Allocation with Inter-Beam Interference in Satellite Communication Systems with a Digital Channelizer,” IEEE Transactions on Wireless Communications, vol. 19, no. 5, pp. 2934 – 2945, May 2020.

[Khan22] B. S. Khan, S. Jangsher, A. Ahmed and A. Al-Dweik, "URLLC and eMBB in 5G Industrial IoT: A Survey," in IEEE Open Journal of the Communications Society, vol. 3, pp. 1134-1163, 2022, doi: 10.1109/OJCOMS.2022.3189013.

[Khosravirand16] S. R. Khosravirad, K. I. Pedersen, L. Mudolo and K. Bakowski, "HARQ Enriched Feedback Design for 5G Technology," 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall), Montreal, QC, Canada, 2016, pp. 1-5, doi: 10.1109/VTCFall.2016.7881032.

[Koopman02] P. Koopman, "32-bit cyclic redundancy codes for Internet applications," Proceedings International Conference on Dependable Systems and Networks, Washington, DC, USA, 2002, pp. 459-468.

[Kurose21] J. F. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach, 8th ed. Boston, MA: Pearson, 2021.

[Liu23] S. Liu, P. Reviriego, A. Ullah, A. Louri and F. Lombardi, "Error-Resilient Data Compression with Tunstall Codes," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 70, no. 5, pp. 1963-1975, May 2023.

[Martinez23] G. Martinez, J. A. Hernandez, P. Reviriego, and P. Reinheimer, “Round trip time (rtt) delay in the internet: Analysis and trends,” IEEE Network, pp. 1–6, 2023.

[Makkena23] Y. C. Makkena et al., "Experience: Implementation of Edge-Cloud for Autonomous Navigation Applications," 2023 15th International Conference on COMmunication Systems & NETworkS (COMSNETS), Bangalore, India, 2023, pp. 579-587, doi: 10.1109/COMSNETS56262.2023.10041370.

[Misa21] C. Misa, R. Durairajan, R. Rejaie and W. Willinger, "Revisiting Network Telemetry in COIN: A Case for Runtime Programmability," in IEEE Network, vol. 35, no. 5, pp. 14-20, September/October 2021.

[Msadaa22] I. C. Msadaa, S. Zairi and A. Dhraief, "Non-Terrestrial Networks in a Nutshell," in IEEE Internet of Things Magazine, vol. 5, no. 2, pp. 168-174, June 2022, doi: 10.1109/IOTM.007.2100121.

[Oliva16] A. de la Oliva, J. A. Hernandez, D. Larrabeiti and A. Azcorra, "An overview of the CPRI specification and its application to C-RAN-based LTE scenarios," in IEEE Communications Magazine, vol. 54, no. 2, pp. 152-159, February 2016, doi: 10.1109/MCOM.2016.7402275.

[Pan19] T. Pan, T. Huang, X. Li, Y. Chen, W. Xue, and Y. Liu, “Opspf: Orbit prediction shortest path first routing for resilient leo satellite networks,” in ICC 2019 - 2019 IEEE International Conference on Communications (ICC), 2019, pp. 1–6.

[Reviriego22] P. Reviriego, J. Martínez, O. Rottenstreich, S. Liu and F. Lombardi, "Remove Minimum (RM): An Error-Tolerant Scheme for Cardinality Estimate by HyperLogLog," in IEEE Transactions on Dependable and Secure Computing, vol. 19, no. 2, pp. 966-977, 1 March-April 2022.

[Reviriego24] P. Reviriego, J. Apple, A. Alonso, O. Ertl and N. Dayan, "Cardinality Estimation Adaptive Cuckoo Filters (CE-ACF): Approximate Membership Check and Distinct Query Count for High-Speed Network Monitoring," in IEEE/ACM Transactions on Networking (in press).

[Richardson21] T. J. Richardson, M. A. Shokrollahi and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," in IEEE Transactions on Information Theory, vol. 47, no. 2, pp. 619-637, Feb 2001.

[Rinaldi20] F. Rinaldi et al., "Non-Terrestrial Networks in 5G & Beyond: A Survey," in IEEE Access, vol. 8, pp. 165178-165200, 2020, doi: 10.1109/ACCESS.2020.3022981.

[Solaija21] M. S. J. Solaija, H. Salman, A. B. Kihero, M. I. Sağlam and H. Arslan, "Generalized Coordinated Multipoint Framework for 5G and Beyond," in IEEE Access, vol. 9, pp. 72499-72515, 2021, doi: 10.1109/ACCESS.2021.3079190.

[Seifert08] R. Seifert, and J. Edwards. The all-new switch book: the complete guide to LAN switching technology. John Wiley & Sons, 2008.

[Tian23] D. Tian, et al. "Efficient protection of polar decoders against Single Event Upsets (SEUs) on user memories." Microelectronics Reliability 149, 2023.

[Vanelli-Coralli20] A. Vanelli-Coralli, A. Guidotti, T. Foggi, G. Colavolpe and G. Montorsi, "5G and Beyond 5G Non-Terrestrial Networks: trends and research challenges," 2020 IEEE 3rd 5G World Forum (5GWF), Bangalore, India, 2020, pp. 163-169, doi: 10.1109/5GWF49715.2020.9221119.

[Vook18] F. W. Vook, A. Ghosh, E. Diarte and M. Murphy, "5G New Radio: Overview and Performance," 2018 52nd Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 2018, pp. 1247-1251, doi: 10.1109/ACSSC.2018.8645228.

[Vukobratovic22] D. Vukobratovic et al., "Edge Machine Learning in 3GPP NB-IoT: Architecture, Applications and Demonstration," 2022 30th European Signal Processing Conference (EUSIPCO), Belgrade, Serbia, 2022, pp. 707-711, doi: 10.23919/EUSIPCO55093.2022.9909793.

[Wang09] H. Wang, C. Ma and L. Zhou, "A Brief Review of Machine Learning and Its Application," 2009 International Conference on Information Engineering and

Computer Science, Wuhan, China, 2009, pp. 1-4, doi: 10.1109/ICIECS.2009.5362936.

[Yang20] H. Yang, A. Alphones, Z. Xiong, D. Niyato, J. Zhao and K. Wu, "Artificial-Intelligence-Enabled Intelligent 6G Networks," in IEEE Network, vol. 34, no. 6, pp. 272-280, November/December 2020, doi: 10.1109/MNET.011.2000195.

[Yang23] K. Yang et al., "SketchINT: Empowering INT With TowerSketch for Per-Flow Per-Switch Measurement," in IEEE Transactions on Parallel and Distributed Systems, vol. 34, no. 11, pp. 2876-2894, Nov. 2023.

[Zhang21] C. Zhang, X. Mu, J. Yuan, H. Li and B. Bai, "Construction of Multi-Rate Quasi-Cyclic LDPC Codes for Satellite Communications," in IEEE Transactions on Communications, vol. 69, no. 11, pp. 7154-7166, Nov. 2021.