Project no.:	101095738		
Project full title:	6G Short range extreme communication IN Entities		
Project Acronym:	6G-SHINE		
Project start date:	01/03/2023	**Duration**	30 months

D4.4 – FINAL RESULTS ON THE MANAGEMENT OF TRAFFIC, COMPUTATIONAL AND SPECTRUM RESOURCES AMONG SUBNETWORKS IN THE SAME ENTITY, AND BETWEEN SUBNETWORKS AND 6G NETWORK

Due date	30/06/2025	**Delivery date**	24/06/2025
Work package	WP4		
Responsible Author(s)	Dimitrios Alanis, Anders Berggren, Baldomero Coll Perales, Pedro Maia de Sant Ana, Filipe Conceicao		
Contributor(s)	Keyvan Aghababaiyan (UMH), Dimitrios Alanis (APPLE), Anders Berggren (SONY), Baldomero Coll Perales (UMH), Javier Gozalvez (UMH), Christian Hofmann (APPLE), Pedro Maia de Sant Ana (BOSCH), Yasser Mestrah (IDE), Filipe Conceicao (IDE),		
Version	V1.0		
Reviewer(s)	Frank Burkhardt (FHG), Davide Dardari (CNIT)		
Dissemination level	Public		

Co-funded by the European Union

Horizon Europe Grant Agreement No. 101095738. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or SNS JU. Neither the European Union nor the granting authority can be held responsible for them.

VERSION AND AMENDMENT HISTORY

Version	Date (MM/DD/YYYY)	Created/Amended by	Changes
0.1	01/21/2025	Christian Hofmann	ToC finalization
0.2	03/21/2025	Dimitrios Alanis	Initial version
0.3	05/05/2025	Dimitrios Alanis	Added general introduction and conclusions
0.4	12/05/2025	Dimitrios Alanis	Draft ready for internal review
0.5	15/05/2025	Frank Burkhardt, Davide Dardari	Internal review completed
0.6	06/06/2025	Dimitrios Alanis	Updates based on reviewers' comments
0.7	10/06/2025	Dimitrios Alanis	Further minor refinements
0.8	17/06/2025	Dimitrios Alanis	Version ready for Company approval
0.9	19/06/2025	Berit H. Christensen	Final proofreading (UK English) and layout check
1.0	24/06/2025	Dimitrios Alanis	Submitted version

TABLE OF CONTENTS

FIGURES

TABLES

ABBREVIATIONS

Abbreviation	Description
3GPP	3rd Generation Partnership Project
4G	4th Generation
5G	5th Generation
5G-ACIA	5G Alliance for Connected Industries and Automation
6G	6th Generation
AD	Autonomous Driving
AF	Application function
AGV	Automated Guided Vehicle
AI	Artificial Intelligence
AMF/AUSF	Access and Mobility Management Function / Authentication Server Function
AMR	Autonomous Mobile Robot
AP	Access Point
AR	Augmented Reality
AS	Access Stratum
B5G	Beyond 5G
BSR	UE Buffer Status Report

CCN	Compute Offload Controlling Node
CCREF	Computing & Communication Resources Exposure Functions
CCRMF	Communication & Computational Resources Management Function
CEV	Crew Exploration Vehicle
Cobots	Collaborative Robots
CompN	6G Network Compute Node
CP	Control Plane
CSI	Channel State Information
D2D	Device to Device
DCS	Distributed Coordination System
DGF	Device Group Function
DN	Data Network
DRB	Data Radio Bearer
E/E	Electrical/Electronic
ECU	Electronic Control Unit
EN	Entity
FCAPS	Fault, Configuration, Accounting, Performance, and Security
gNB	gNodeB (i.e., the functional equivalent of a base-station)
GW	Gateway Role
HARQ	Hybrid Automatic Repeat request
HC	Element with High Capabilities
HPCU	High-Performance Computing Unit
JFI	Jain Fairness Index
IAB	Integrated Access Backhaul
IVN	In-Vehicle Network
KDF	Key Derivation Function
KPIs	Key Performance Indicators
LC	Element with Low Capabilities
LCh	Logical Channel
LU	Location Updates
LiDAR	Light Detection and Ranging
MAC	Medium Access Control

MgtN	Management Node
N3SMF	Non-3GPP Subnetwork Management Function
NEF	Network Exposure Function
NIF	Network Intelligence Functions
NoN	Network of Networks
NSA	Non-Standalone
NWDAF	Network Data Analytics Function
Near-RT	Near Real-Time
Non-RT	Non-Real-Time
OFDMA	Orthogonal Frequency Division Multiple Access
OFF	Compute Offloading
O-RAN	Open Radio Access Network
PDCP	Packet Data Convergence Protocol
PHY	Physical Layer
ProSe	Proximity Services
QoS	Quality of Service
RAN	Radio Access Network
RLC	Radio Link Control
RIS	Reconfigurable Intelligent Surface
RNAU	Radio Network Area Update
ROI	Region of Interest
RRM	Radio Resource Management
SAF	Sensing Analytics Function
SCF	Sensing Coordination Function
SCS	Subcarrier Spacing
SDAP	Service Data Adaptation Protocol
SL	Sidelink
SMO	Service Management and Orchestration
SN	In-X Wireless Subnetwork
SN-TP	Subnetwork Tunnelling Protocol
SNE	Subnetwork Element
SNM	Subnetwork Management

TA	Tracking Area
TAU	Tracking Area Update
TB	Transport Block
UE	User Equipment
URLLC	Ultra-Reliable Low Latency Communication
VR	Virtual Reality
WLAN	Wireless Local Area Network
XR	Extended Reality

EXECUTIVE SUMMARY

This deliverable reports the finalized activities carried out in Task 4.2 “Resource management within subnetwork entity and towards the 6G network”. It builds upon the initial activities presented in deliverable D4.2 as well as the architectural framework of deliverable D2.4. Initially in Chapter 2, the connection between the novel methods proposed in this report and the use cases, as they were defined in the deliverables D2.1 and D2.2, is elaborated.

Moving on to Chapter 3, final studies of routing of data and control signalling within subnetworks in the same entity are presented corresponding to Task 4.2a. At first, detailed processes of the novel architectural enablers are presented in Section 3.1. These processes include security, configuration and data multiplexing of the so-called Non-Standalone UEs, followed by UE-centric authentication and local IP routing procedures. Treatises related to subnetwork formation and (re-)selection as well as mobility within subnetworks, across subnetworks and with the parent network are presented in Section 3.2. Subsequently, processes related to coordination within and across subnetworks are presented in Section 3.3. In this context, the location update of a group of devices belonging to the same subnetwork is offloaded to the subnetwork management node. Additionally, a framework that offloads the L3 measurements of lower capability subnetwork device to higher capability ones is introduced.

Subnetwork Quality-of-Service (QoS) aspects with a focus on time alignment of multi-modal data are investigated in Section 3.4. Two alternative approaches are presented focusing on the consumer use case category: a network-controlled approach that enhances the current 3GPP Sidelink Relay framework, as well as a UE-centric approach that offloads the time alignment control at the management node. Besides multi-modal data time alignment, the problem of UE scheduling within the subnetwork is also investigated and new protocols which are deployed at the management node are introduced.

Apart from network functionality offloading, the problem of compute offloading is also investigated in Chapter 4, mapped to the work of Task T4.2b. The processes enabling both local, within the subnetwork, as well as decentralised compute offloading are introduced in Section 4.1. Additionally, an enhancement to 3GPP QoS framework is introduced in Section 4.2 to enable reliable compute offloading with QoS guarantees. For the safety-critical in-vehicle use case category, a deterministic task offloading and resource allocation scheme for the integrated management of communication and computing resources across the IoT-edge-cloud continuum is presented in Section 4.3. Focusing on the same use case category, a deterministic task scheduling scheme for in-vehicle networks (IVNs) is presented in Section 4.4 with its potential to leverage the capabilities of in-vehicle zonal E/E architectures with centralized computing. Finally, solutions are presented in Section 4.5 that enable subnetworks to select the best site to instantiate a terminal service, considering service-specific requirements at both connectivity and computing levels.

The problem of dynamic spectrum sharing is also investigated in Chapter 5 according to Task T4.2c. At first, a review of the spectrum sharing regulations across countries is presented in Section 5.1. More specifically, a comparison of licensed and license-exempt spectrum policies is presented, followed by an evaluation of sharing mechanisms. Subsequently, the compliance and enforcement approaches are reviewed and the emerging trends in spectrum regulation are identified. An analysis of the implications for future 6G spectrum policy is then made. Additionally, a novel protocol for flexible access of licensed

resources is proposed in Section 5.2, where the parent network functionality of dynamically assigning licensed resources to their registered subnetworks is introduced.

To conclude this deliverable which constitutes the final study on Task T4.2, an overall discussion regarding the 6G-SHINE project objectives and targets is provided in Chapter 6.

1 INTRODUCTION

This document presents the final studies on the management of resources in subnetworks (SN) located in close vicinity as well as of resources shared with the overlay 6G network (NW). The document addresses Objective 6 of the 6G-SHINE project "Develop new methods for integration of subnetworks in the 6G architecture and efficient orchestration of radio and computational resources among subnetworks and wider network" and it is directly related to 6G-SHINE Task T4.2. The term "resources" is generalized in the context of 6G NWs by extension to the SNs, as it refers to not only spectral resources for transmission purposes, but also to resources that enable functional and computational offloading. Within this scope, there is an inherent need for coordination mechanisms with the SN as well as across SNs to achieve the best possible performance in terms of the KPIs defined in [1] and [2], such as data rate and latency. Additionally, the studies presented in this document constitute a continuation of the SN as proposed in [3] and of the initial studies presented in [4].

1.1 RESOURCE MANAGEMENT WITHIN A SUBNETWORK ENTITY AND TOWARDS THE 6G NETWORK

The use cases anticipated in 6G NWs [5] assume an inherent densification of the nodes, which in turn impose a significant overhead on both the Core Network (CN) and the Radio Access Network (RAN) sides. In the current cellular architecture, individual direct connections shall be established between each of the UEs and the RAN, thus yielding significant control and signalling overheads at the network side. To address this scalability issue that arises with the nodes' densification, the concept of the *networks of networks* (NoN)[5] shall be applied, regarded as one of the pillars of 6G [6]. In this context, the nodes are organised into hierarchical nested smaller networks, which are referred to as SNs. This hierarchical structure provides the possibility of reducing control and routing overhead at the RAN side by applying the divide-and-conquer principle. More specifically, the SN nodes are served, coordinated and controlled by special UE nodes, referred to as *Management Node* (MgtN) [2] [3]. This type of node provides the nodes connectivity to other SNs or to the overlay NW. This hierarchical organisation of the nodes, as portrayed in Figure 1, provides opportunities for local coordination at SN level leveraging from the close vicinity in which the nodes are located. This in turn yields potential control and data overhead reduction, achieved by offloading network functionality from the parent 6G NW to the SNs.

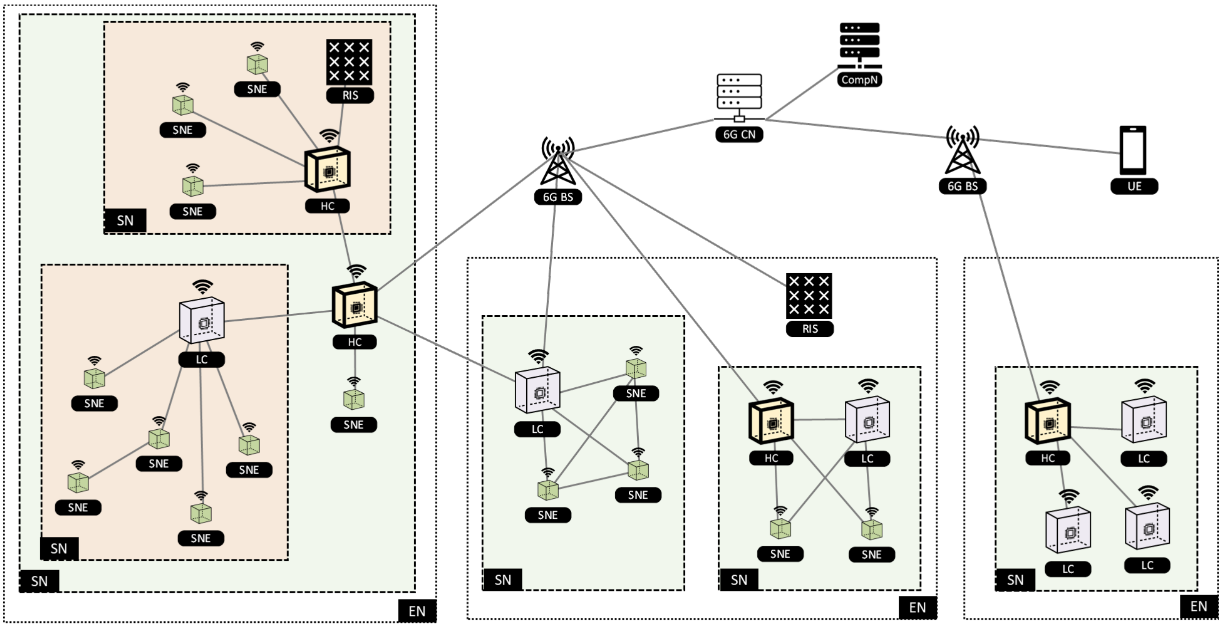


Figure 1 SN Reference architecture [3].

As pointed out in the preliminary results of [4], a meticulous SN protocol design is required in order to achieve the degree of user autonomy and privacy required by the use cases of [2], while at the same time maintaining a seamless integration into the parent network. In this context, the concept of virtual connections between the UE and the parent 6G BS was introduced in [4] and the SN architecture was introduced in [3]. These virtual connections allow the UE to be reachable by the 6G BS, when joining a specific subnetwork while catering for maintaining connectivity to the 6G NW in the cases, where the UE joins and leaves an SN. In this context, distributed SN Control Plane (snCP) and SN User Plane (snUP) were proposed that allowed flexible deployments with UEs of diverse capabilities. Most notably the concept of the so-called Non-Standalone UE (NSA-UE) was introduced, for specific UEs that can achieve direct connection to the parent 6G BS only with the aid of an assisting UE, which is referred to as standalone UE (SA-UE). Additionally, the protocols for achieving data routing and control within the SN as well as with the parent 6G BS were introduced in the form of the SN Routing Protocol (SN-RP) and SN Tunnelling Protocol (SN-TP), respectively. Additionally, a coordination framework was proposed [4] aiming at providing deterministic service level provisioning in hybrid wireless and wired in-vehicle SNs, which are inherently static deployments without any node mobility. A review on the 3GPP framework for QoS in Sidelink (SL) relay deployments was also made in [4] with a glance at data multi-modality focusing on XR-related applications.

Besides enhancements for communications, the novel use cases of [2] demand the enablement of compute offloading, leading to converged computation and communications networks. Compute offloading unlocks an additional type of SN resources for harnessing, allowing the deployment of less complex nodes. In this context, a preliminary framework-based approach was presented in [4] relying upon aspects such as the traffic characterisation, the architecture definition, and service characteristics. Additionally, the roles required for enabling compute offloading, have been introduced in [4].

The coordination of SNs with their neighbouring SNs as well as with the parent 6G NW raises an important issue, namely that of the subnetwork's and overlay 6G network's coexistence in terms of spectral resources. For this reason, efficient dynamic spectrum sharing mechanisms should be designed. In the preliminary study of [4], a survey on the advantages and drawbacks of licensed versus unlicensed spectrum was made, while taking into consideration the regulatory constraints across countries as well as the intricacies of the use cases investigated in [2].

1.2 SUMMARY OF THE PROPOSED SOLUTIONS

Based on the work of [4], this report continues the investigation into novel protocols and processes for SNs in order to enable the deployment of the use cases of [2], converging computation with communication. Note that a direct mapping of the solutions presented in this report is made in Chapter 2.

In Chapter 3, the solutions designed for addressing the routing of data and control signalling within the SN, across SNs and with the parent 6G NW are presented. This chapter is directly mapped to 6G-SHINE's task T4.2a. In this context, some novel procedures related to SN architectural enhancements are made. Configuration, data multiplexing and security mechanisms are introduced for enabling the deployment of NSA UEs. UE-centric authentication processes are introduced with an aim of enhancing UE autonomy by bringing the authentication process closer to the UEs by not involving the CN. Autonomy from the CN is further enhanced by keeping data of source and destinations nodes of the same or neighbouring SNs within the SN boundaries. This is achieved by a novel IP connectivity protocol which enhances the SN-RP of [4].

Besides architectural enhancements, novel SN formation, registration and mobility procedures are also introduced in Chapter 3. In terms of SN formation, a decentralized process without any parent CN involvement for selecting which nodes become MgtNs has been proposed. NW registration processes are then introduced so that the MgtNs make the NW aware of their SN, so that the NW grants resources to the SN and associates the SN UEs with the SN. As far as SN mobility is concerned, UE-centric processes for SN (re-)selection are introduced, enabling devices to connect to the most appropriate SN according to their needs. The element of coordination is introduced for mobility processes, where SN assists not only on SN re-selection but also on parent cell selection, by the MgtN gathering and transmitting side information for re-selection decisions to it served UEs.

Apart from mobility, coordination is also introduced in certain CP functions as well as for L3 measurements offloading. As for CP function offloading, the problem of location updates is investigated, and coordination mechanisms are proposed for performing these updates at the MgtN on behalf of the UEs. As for Layer 3 (L3) measurements, a coordinated framework is proposed, where SN lower capabilities UEs can offload their L3 measurements to higher capability UEs either for power consumption reduction or for enhancing the accuracy of their L3 measurements.

Additionally, aspects of Quality-of-Service (QoS) within SNs are outlined in Chapter 3. More specifically, data multi-modality has been investigated, i.e. multiple devices attached to the users interact with each other within a SN and/or across SNs. This creates multi-modal flows where the packet delivery needs to be synchronised across devices. Two approaches are followed. In the first approach, time alignment is accomplished by adding information in the existing 3GPP framework of the related flows, and of the interrelated packets in the related flows, in the packet headers. Thereby the relays and devices can synchronize the dataflows and still maintain a relevant packet delay budget which improves the performance and capacity of the network. As for the second approach, a more SN-centric method is followed, where the MgtN is equipped with a Device Group Function (DGF), which in turn is responsible for performing time alignment on the multi-modal streams. Additionally, the issue of UE scheduling is investigated. In this context, a novel per-UE Buffer Status Report (BSR) is introduced that enables the MgtN to transmit to the parent NW the individual buffer status levels of each of its serving UEs in the same message. The new per-UE BSR enables two UE scheduling schemes within the SN: where data is buffered at the MgtN and where extra information about the UEs' scheduling latencies is considered eliminating the need for buffering.

Moving on to Chapter 4, the protocols related to compute offloading for dynamic topologies are presented. This chapter is directly mapped to 6G-SHINE task 4.2b. Building upon the SN compute offloading roles proposed in the preliminary results of [4] as well as in [7], a new set of roles are introduced, namely those of the managing and assisting Compute offload Control Node (CCN). With the aid of these new roles, new decentralized processes are presented for determining the CCN nodes, which in turn match Offloading Nodes (ON) to Computing Nodes (CompN) and control the overall compute offloading process. The proposed framework is not only restricted to compute offloading within a single SN but is it also extended to enable compute offloading to neighbouring SNs or to more remote SNs using the communication infrastructure from the parent 6G NW. This extended framework is referred to as decentralized compute offload.

For the sake of achieving a converged computation and communication SN, the existing QoS framework needs to be revisited to include computation aspects. This is necessary for fully utilizing the computation offloading capability in the SN architecture and to support computation requests with different resources and performance requirements. For this reason, a novel Quality of Computation Service (QoCS) framework is introduced in Chapter 4, supporting both communication and computation within a SN, between SNs, and between the SN and the parent 6G NW. In this context, novel SN QoCS parameters and characteristics to fulfil required computation requirements along with the high-level procedures to support SN QoCS for local SN and decentralised compute offload are presented.

The in-vehicle uses cases of [2] are inherently static deployments with tight service level provisioning constraints. For this reason, the respective protocols need to be tailored to take these constraints into account. In this context, two studies are presented. At first, a deterministic task offloading and resource allocation scheme for the integrated management of communication and computing resources across the IoT-edge-cloud continuum is presented. Subsequently, a novel deterministic task scheduling scheme for in-vehicle networks (IVNs) is presented with its potential to leverage the capabilities of in-vehicle zonal E/E architectures with centralized computing.

Moving on to Chapter 6, the studies related to dynamic spectrum sharing are presented. At first, review of the spectrum sharing regulations across countries is made. More specifically, a comparison of licensed and license-exempt spectrum policies in the EU, China, and the US is presented, followed by an evaluation of sharing mechanisms, such as EU’s LSA, US CBRS models. Subsequently, a review of the compliance and enforcement approaches is made and the emerging trends in spectrum regulation are identified. An analysis of the implications for future 6G spectrum policy is then made. Additionally, a novel protocol for flexible access of licensed resources is proposed. The concept of dynamic resource pools is introduced, which the NW assigns to SNs depending on their traffic needs.

Last but not least, an overall discussion regarding the 6G-SHINE project objectives and targets is presented in Chapter 6, followed by the conclusion in Chapter 7.

2 ENABLEMENT OF RELEVANT 6G-SHINE USE CASES

This chapter provides an in-depth correlation between the use cases introduced in Deliverable D2.2, "Refined Definition of Scenarios, Use Cases, and Service Requirements for In-X Subnetworks" [2], and the technical innovations presented in this document.

Deliverable D2.2 presents a broad set of in-X subnetwork use cases spanning multiple domains including consumer, industrial and vehicular use cases. The consumer use case focuses on making immersive media experiences more widely accessible, moving beyond high-end setups to mainstream use through wireless technologies and smartphones. Applications include virtual product visualization, gaming, and immersive entertainment. The industrial use case centres around the transition to Industry 4.0, which integrates cyber-physical systems, IIoT, and cloud computing to create intelligent, interconnected manufacturing environments. The vehicular use case addresses the increasing communication demands within Connected and Automated Vehicles (CAVs) and Electric Vehicles (EVs), where sensors, actuators, control units, and computing systems must exchange data reliably and safely.

Most of the use cases in different domains are relevant to the technical developments covered in this deliverable. The first use case from consumer domain is "Immersive Education" (Figure 2). Immersive Education aims to elevate the interactive learning experience between students and teachers by leveraging advanced media content and related technologies. It extends the learning environment beyond the conventional classroom, enabling students to engage with course material more intuitively and effectively. By accommodating diverse learning styles through rich XR experiences and varied stimuli, it promotes consistent learning outcomes. Additionally, this approach fosters meaningful interaction among students and supports the seamless inclusion of remote or virtual learners.

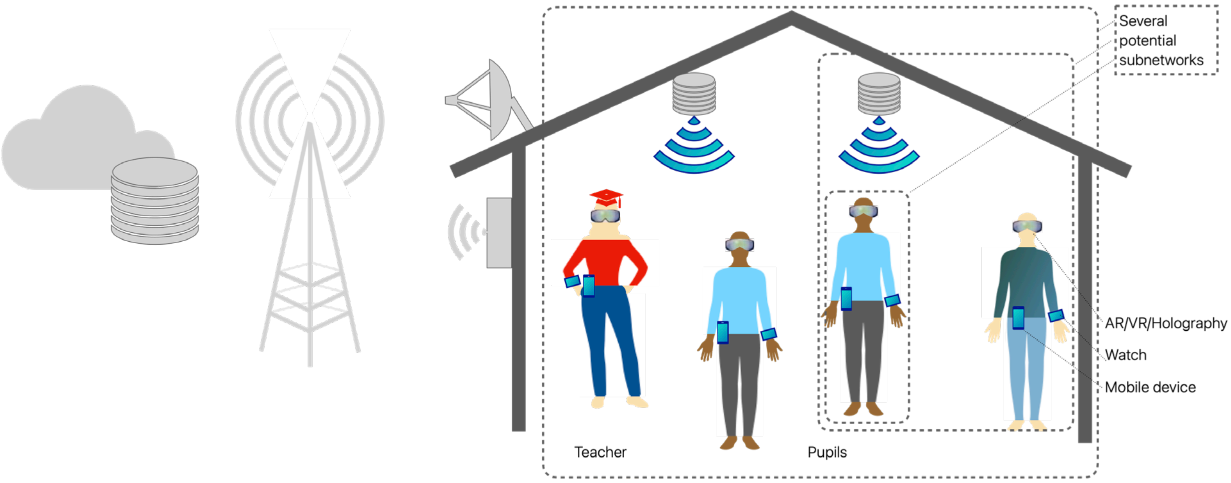


Figure 2 Illustration of immersive education use case

The technical solutions proposed in this document such as subnetwork architecture and the introduction of new device types, a framework for subnetwork configuration and the authentication of new students and teachers, subnetwork formation and mobility support to enable dynamic participation, coordination mechanisms within subnetworks, and protocols for computational offloading to edge nodes and cloud servers are all well-aligned with the requirements of the Immersive Education use case. Additionally, the Quality of Compute Service framework supports diverse QoS flows, while flexible local routing enables efficient task distribution. Capabilities such as dynamic spectrum sharing for adaptable access and granted subnetwork resource sharing for spectrum allocation further enhance performance. Collectively, these advancements support the delivery of responsive, adaptive, and high-quality educational experiences across both physical and virtual environments.

Another closely related use case from consumer domain is “Augmented Reality (AR) Navigation” (Figure 3). This scenario explores the integration of AR navigation with AI/ML-powered digital assistance, primarily within urban environments. Typically delivered through AR glasses, the system overlays relevant, real-time information onto the user’s view of the physical world. These AR devices are often equipped with various components, including sensors, a microphone, a camera, a speaker, and a communication module. Functionally, the AR device captures environmental and user-specific inputs, transmits this data to a server or processing node, receives the processed information, and delivers it back to the user through visual or audio output. An AI/ML server supports this system by analysing multiple input streams to generate personalized, context-aware information, thereby enhancing user navigation and situational awareness.

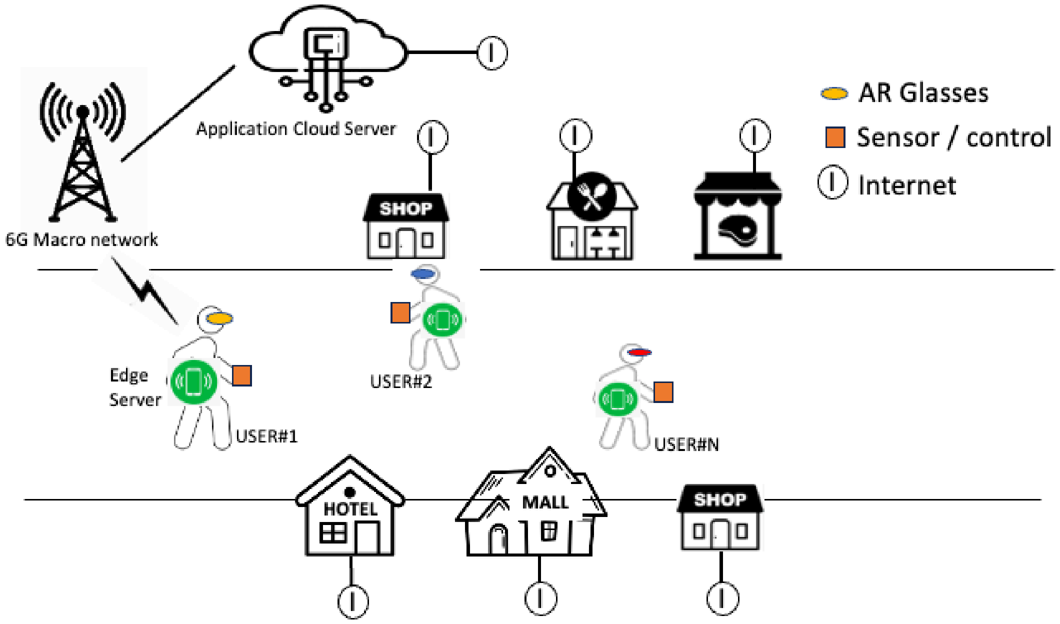


Figure 3 Illustration of AR Navigation

The technical solutions proposed in this document such as subnetwork formation and mobility support, coordination mechanisms within subnetworks, protocols for computational offloading to edge nodes and cloud servers, and the Quality of Compute Service framework for supporting differentiated QoS flows are all directly applicable to the AR Navigation use case. These protocols enable the offloading of AI/ML processing tasks, such as real-time scene analysis and user-specific guidance, from resource-limited AR devices to more capable edge/cloud infrastructures. This ensures low-latency, context-aware assistance that adapts dynamically to the user's surroundings. Furthermore, joint task and communication scheduling supports dependable service provisioning, while dynamic spectrum sharing and granted subnetwork resource sharing enhance connectivity and spectrum efficiency. Collectively, these capabilities ensure the delivery of reliable, responsive, and personalized AR experiences in complex and bandwidth-constrained urban environments.

Another example in the consumer category is indoor gaming. This use case is characterized by stringent Quality of Service (QoS) requirements, including low latency, high data rates, and reliable communication. The use case includes, e.g., AR/VR headset, audio and haptic sensors. To get an immersive experience for all involved participants in the game, the output from sensors and delivery of audio and video input needs to be delivered in a synchronized manner. This puts stringent requirements on any QoS framework responsible for handling the gaming use case, but suitable to be handled in a subnetwork reducing the interaction with parent 6G network in some scenarios.

Furthermore, this deliverable expands its scope to include industrial use cases. A particularly relevant scenario aligned with the technical innovations presented here is “Subnetworks Swarms: Subnetwork Co-existence in Factory Hall” (Figure 4). In modern manufacturing environments especially within the electronics and automotive industries tasks are increasingly distributed among swarms of smaller, specialized robots. Each robot is configured to perform specific operations, and when functioning collectively, these robotic swarms can achieve levels of efficiency and flexibility that often surpass traditional assembly lines. Central to this coordinated effort is the principle of collaborative problem-solving, where each robot not only executes its assigned tasks but also communicates and shares information with other members of the swarm. In this context, the concept of subnetworks is elevated to a higher hierarchical level, referring to the dynamic, interconnected network of collaborating robotic units.

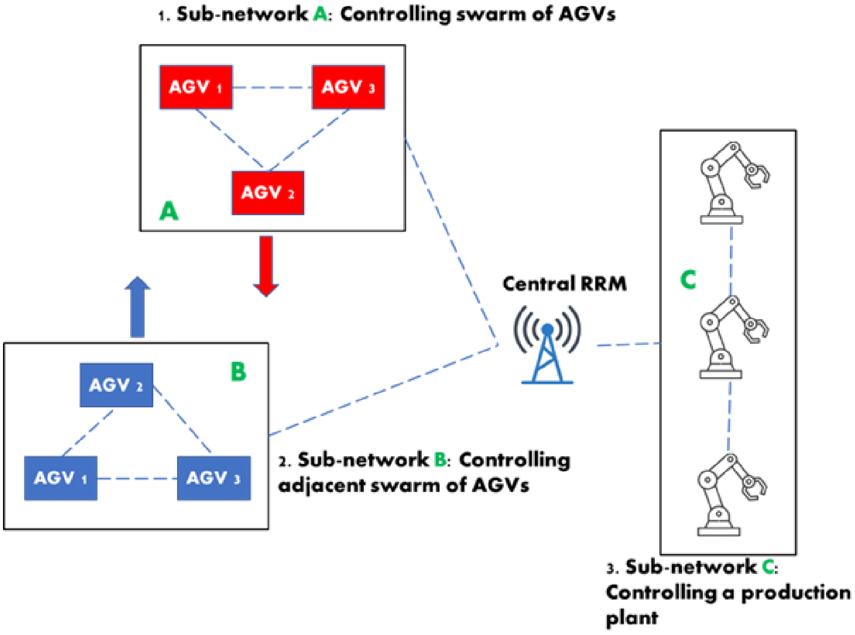

1. Sub-network A: Controlling swarm of AGVs
AGV 1
AGV 3
AGV 2
A
Central RRM
C
B
AGV 2
AGV 1
AGV 3
2. Sub-network B: Controlling adjacent swarm of AGVs
3. Sub-network C: Controlling a production plant

Figure 4 Illustration of the Subnetwork Co-existence in Factory Hall Use Case

The technical solutions proposed in this document such as subnetwork architecture, dynamic subnetwork formation and mobility support, coordination mechanisms within and across subnetworks, and protocols for computational offloading are highly applicable to the Subnetworks Swarms use case. In industrial settings where multiple robotic units operate concurrently, these protocols enable the flexible and efficient organization of robot swarms into logical subnetworks that can adapt to evolving tasks and spatial arrangements. Offloading computationally intensive operations, such as collective decision-making and real-time sensor data processing, to edge nodes or centralized servers enhances the responsiveness and scalability of the system. The Quality of Compute Service framework ensures that each robot receives appropriate compute and communication resources based on its role and task urgency. Moreover, dynamic spectrum sharing and granted subnetwork resource sharing are critical for minimizing interference and optimizing wireless communication among co-located subnetworks operating simultaneously within a factory hall. These capabilities collectively support resilient, high-performance robotic collaboration in complex industrial environments.

Furthermore, this deliverable extends its scope to encompass vehicular use cases. A key example is the "Collaborative Wireless Zone ECUs" use case (Figure 5). This scenario focuses on automotive systems and applications that benefit from the collaboration or offloading of functions, sensors, and actuators distributed across multiple zones in-vehicle E/E architecture. Each zone is equipped with a wireless zone ECU, which integrates various sensors and actuators to support a wide range of automotive functions across different vehicle domains. This distributed architecture enables intelligent coordination between in-vehicle components, improving efficiency, scalability, and system responsiveness. Collaboration among wireless zone ECUs is especially critical for advanced vehicular functionalities that require real-time data exchange and distributed processing.

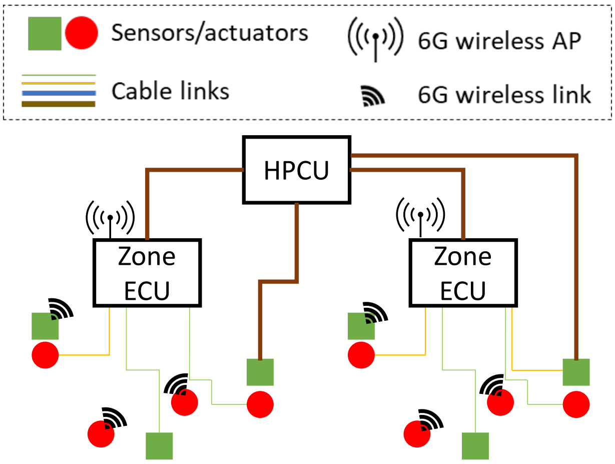


Figure 5 Collaborative wireless zone ECUs use case

The technical protocols and mechanisms outlined in this document such as computational offloading procedures, the Quality of Compute Service framework, and flexible local routing within subnetworks for task distribution are directly aligned with the requirements of the Collaborative Wireless Zone ECUs use case. These capabilities enable seamless coordination and dynamic offloading of processing tasks among distributed in-vehicle ECUs, ensuring efficient resource utilization and low-latency performance across multiple vehicle domains. By supporting real-time data exchange and adaptive task management, these mechanisms enhance the scalability, reliability, and responsiveness of advanced automotive functions within modern E/E architectures.

Another relevant vehicular use case addressed by this deliverable is "Virtual ECUs: In-vehicle Sensor Data and Functions Processing at the 6G Network Edge" (Figure 6). This use case focuses on integrating the in-vehicle network with the broader 6G parent network, in line with the 6G "network of networks" paradigm. The primary goal is to extend the in-vehicle embedded computing capabilities to the edge or cloud, enabling seamless and dynamic collaboration between the vehicle, the network, and cloud infrastructure. By leveraging this integration, the system supports opportunistic offloading and vehicle-network-cloud cooperation to enhance advanced automotive functionalities, particularly those critical for autonomous driving (AD) and the continuous evolution of intelligent vehicles. In this context, the edge or cloud acts as a virtual ECU (or HPCU), elastically extending the vehicle's processing and computational capabilities via the 6G network. For instance, sensor data collected from zones managed by wireless zone ECUs can be offloaded to the edge or cloud for real-time processing, along with computationally intensive tasks such as machine learning inference.

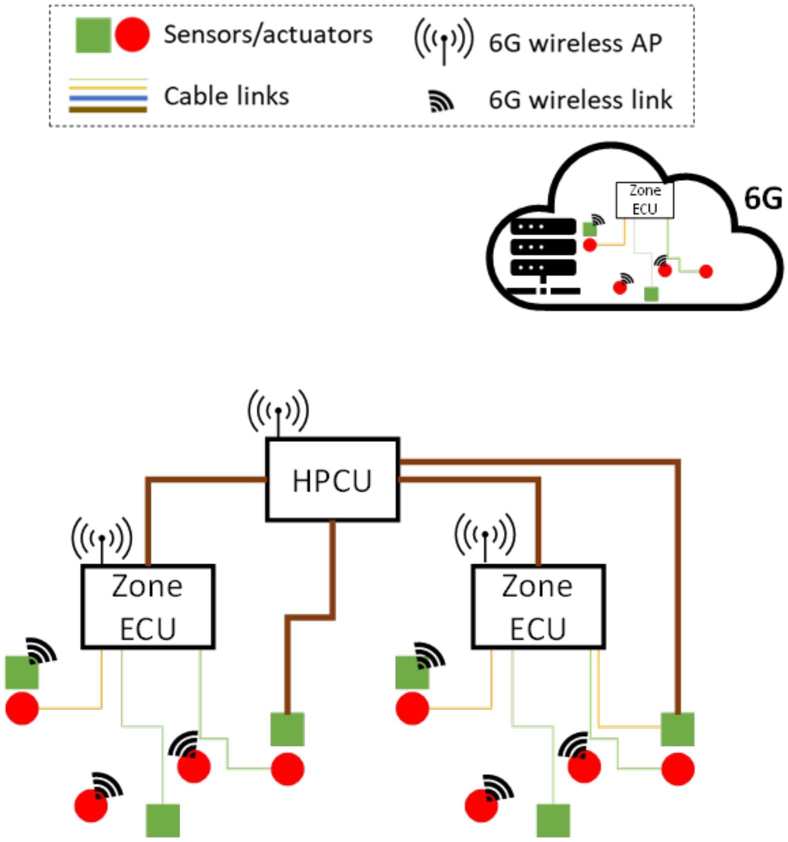


Figure 6 Integration of the 6G in-vehicle network with the 6G parent network.

The technical protocols and mechanisms outlined in this deliverable such as the Quality of Compute Service framework are highly applicable to this use case. By enabling fine-grained resource allocation and dynamic QoS management, this framework supports the elastic extension of in-vehicle computing to edge and cloud infrastructures. Furthermore, advanced capabilities such as joint task and communication scheduling ensure dependable service provisioning, while dynamic spectrum sharing and granted subnetwork resource sharing are critical for maintaining high reliability, low latency, and scalable performance. These technical enablers are essential for realizing seamless vehicle-network-cloud cooperation, particularly in support of data-intensive and time-critical functions required for autonomous driving and next-generation intelligent vehicle systems.

The table below illustrates how the use cases relate to the technologies of this document.

Table 1 Use cases

Use Cases	Technologies of this document
Immersive Education	Subnetwork Architecture and new device types (Section 3.1) Subnetwork Formation and Mobility (Section 3.2) Coordination within Subnetworks (Section 3.3) Protocols and Procedures for Computational Offloading (Section 4.1) Quality of Compute Service (QoCS) Framework for Subnetworks (Section 4.2) Dynamic Spectrum Sharing and Regulation (Section 5.1) Granted Subnetwork Resource Sharing (Section 5.2)

Augmented Reality (AR) Navigation	Subnetwork Architecture and New Device Types (Section 3.1) Subnetwork Formation and Mobility (Section 3.2) Coordination within Subnetworks (Section 3.3) Protocols and Procedures for Computational Offloading (Section 4.1) Joint Task and Communication Scheduling for Dependable Service Level Provisioning (Section 4.3) Compute Aware Traffic Steering with Mobility Considerations (Section 4.5) Dynamic Spectrum Sharing and regulation (Section 5.1) Granted Subnetwork Resource Sharing (Section 5.2)
Subnetworks Swarms	Subnetwork Architecture and new device types (Section 3.1) Coordination within Subnetworks (Section 3.3) Joint Task and Communication Scheduling for Dependable Service Level Provisioning (Section 4.3) Dynamic Spectrum Sharing and Regulation (Section 5.1) Granted Subnetwork Resource Sharing (Section 5.2)
Collaborative Wireless Zone ECUs	Protocols and Procedures for Computational Offloading (Section 4.1) Quality of Compute Service (QoCS) Framework for Subnetworks (Section 4.2) Flexible Local Routing in Subnetwork for Task Offloading (Section 4.4) Dynamic Spectrum Sharing and Regulation (Section 5.1)
Virtual ECUs	Quality of Compute Service (QoCS) Framework for Subnetworks (Section 4.2) Joint Task and Communication Scheduling for Dependable Service Level Provisioning (Section 4.3) Dynamic Spectrum Sharing and Regulation (Section 5.1)

3 ROUTING OF DATA AND CONTROL SIGNALLING WITHIN SUBNETWORKS IN THE SAME ENTITY

This chapter constitutes the continuation of the work of [4] on routing procedures for data and control signalling within and across subnetworks. Naturally, this chapter is directly mapped to 6G-SHINE task 4.2a. More specifically in this Section, enhancements in the architecture of the SNs in conjunction with procedural enhancements to allow a new type of Non-Standalone (NSA) devices are presented in Section3.1. Subsequently, SN formation and mobility procedures are presented in Section 3.2 followed by processes enabling SN coordination in Section 3.3. Finally, solutions on Quality of Service (QoS) aspects, such as multi-modality, scheduling and resource allocation, in the context of SNs are advocated in Section 3.4.

3.1 SUBNETWORK ARCHITECTURE AND NEW DEVICE TYPES

3.1.1 Introduction

This section focuses on new SN architectural elements. As far as NSA devices are concerned, a brief introduction on their necessity and their deployment has been provided in [4]. In Section 3.1.2, a detailed description is given on how these NSA devices are configured as well as how their communication with the 6G-BS is end-to-end secured. Subsequently, the problem of device-to-device authentication without any involvement of the Core Network (CN) is addressed in Section 3.1.3. Finally, enhancements on how to enable local IP connectivity are presented in Section 3.1.4.

3.1.2 Procedural Enhancements for NSA UEs

In the context of dynamic topologies and flexible roles for subnetwork (SN) nodes, D4.2 [4] introduced a new category of Low Capability (LC) devices in the SN, the so-called Non-Standalone (NSA) devices. As highlighted in Figure 7, the NSA devices may require direct 6G Base Station (BS) connections. This is crucial for eliminating unnecessary hops and delays, to fulfil the low-latency requirements of timing/time sensitive use cases such as the immersive education [2]. Although the NSA devices are incapable of establishing these direct links with the 6G BS due to complexity and power constraints, an HC within the SN can assist in establishing the links on their behalf.

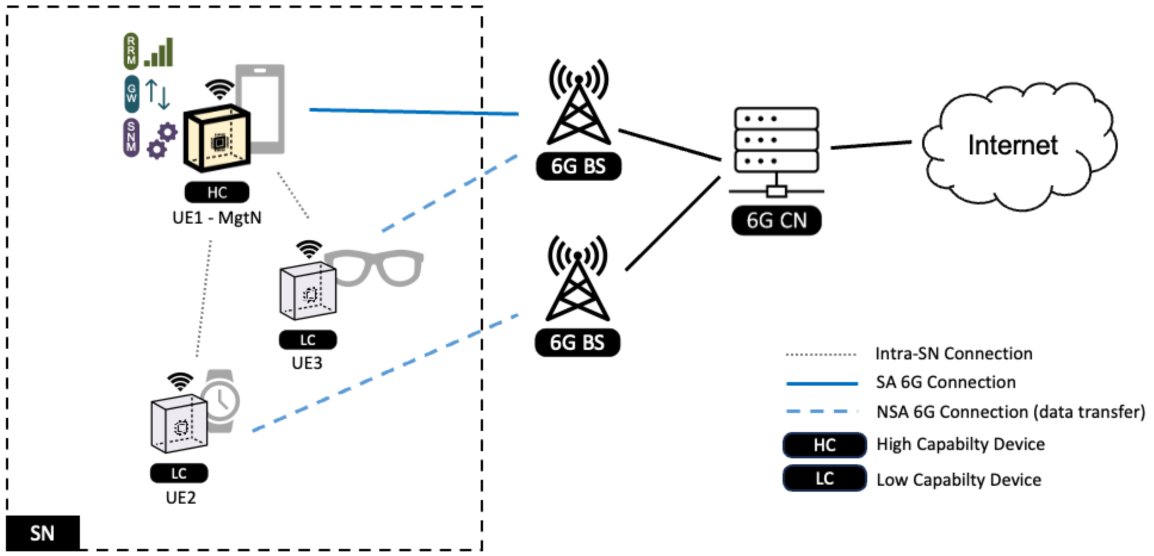


Figure 7 Subnetwork enabling Non-Standalone LC devices for direct 6G connections [4]

To form the SN, mutual authentication between the devices and the MgtN must take place, so that the devices establish secure links with the MgtN. After the SN formation, the 6G system requires procedural enhancements to enable the creation of the proposed direct links towards NSA LC devices, since those are not capable of establishing the links on their own.

3.1.2.1 Configuration of NSA LC devices

The new establishment and configuration scheme involves the NSA LC device (which from now on will be referred to as NSA-UE), the MgtN acting as SA, and the corresponding BS. The envisioned high-level configuration architecture is shown in Figure 8.

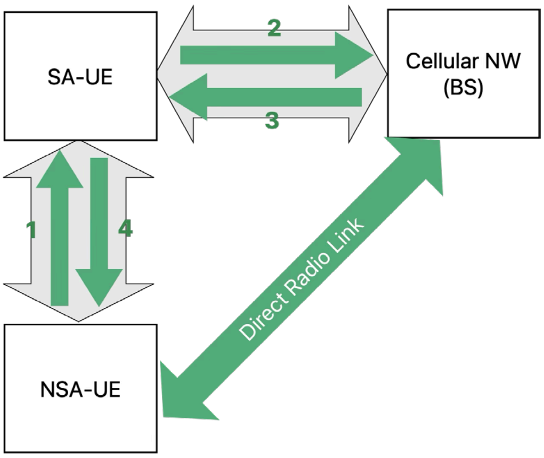


Figure 8 High-level configuration architecture for direct links to NSA LC devices

Figure 9 High-level message flow for direct link establishment

Figure 9 depicts the detailed message flow with the necessary steps for establishment of a direct link between NSA-UE and 6G Network (NW). In order to receive an initial cellular configuration, the NSA-UE shall inform the MgtN (SA-UE) that a direct cellular link is required (corresponding to Step 1 of Figure 8). Based on that, the MgtN may optionally perform a connection establishment procedure to create a communication link towards the NW in order to request direct cellular link for the NSA-UE including necessary radio capabilities and measurements relevant for the NSA-UE (see also Step 2 of Figure 8). The NW shall process the received request along with the received parameters characterizing the NSA-UE radio capabilities and link quality, prepare the NSA-UE cellular configuration required for the direct link and provide it back to MgtN, which corresponds to Step 3 of Figure 8. Finally, the MgtN shall process and forward that configuration in support of the NSA-UE, which is a lower capability device that may have limited functionality to do so (see Step 4 in Figure 8). The NSA-UE shall apply the received configuration and start the UL/DL data transfer via the direct radio link with the 6G BS and the MgtN may enter power saving mode. After this initial setup, the cellular NW may send reconfiguration messages directly to the NSA-UE to further modify the NSA-UE cellular configuration. When data transfer is completed, the Cellular NW may release the connection and accordingly the data session for NSA-UE would also be terminated.

3.1.2.2 User Plane Enhancements for NSA LC devices

A direct link establishment as described in Section 3.1.2.1 requires additional modifications of the User Plane (UP) handling, especially at the NW side. In the absence of a direct link, there are different options regarding how the NSA-UE data is carried to the MgtN, Figure 10 depicts the case where a “PDU Session 1” is carrying both SA-UE traffic and NSA-UE traffic, as well as the case when there is a separate “PDU Session 2” only carrying NSA-UE traffic. Upon establishment of a direct link, the NSA-UE can directly use the newly established direct link for UL traffic, while for the DL traffic the data needs to be re-routed via the direct link towards the NSA-UE. This implies configuration changes at the BS side. Consequently, the NW needs to be informed which PDU sessions or which parts of a PDU session should be routed via the direct link towards the NSA-UE. In case the SA-UE and the NSA-UE are served by the same BS, the SN may provide the BS with QoS flow filters to separate the NSA QoS flows (PDU Session 1) or an indication to route the whole PDU session traffic (PDU Session 2). If the SA-UE and the NSA-UE are served by different BSs, routing of NSA-UE’s traffic should be changed at the UPF, and different paths within the CN need to be enabled to deliver NSA data to the specific BS serving the NSA-UE.

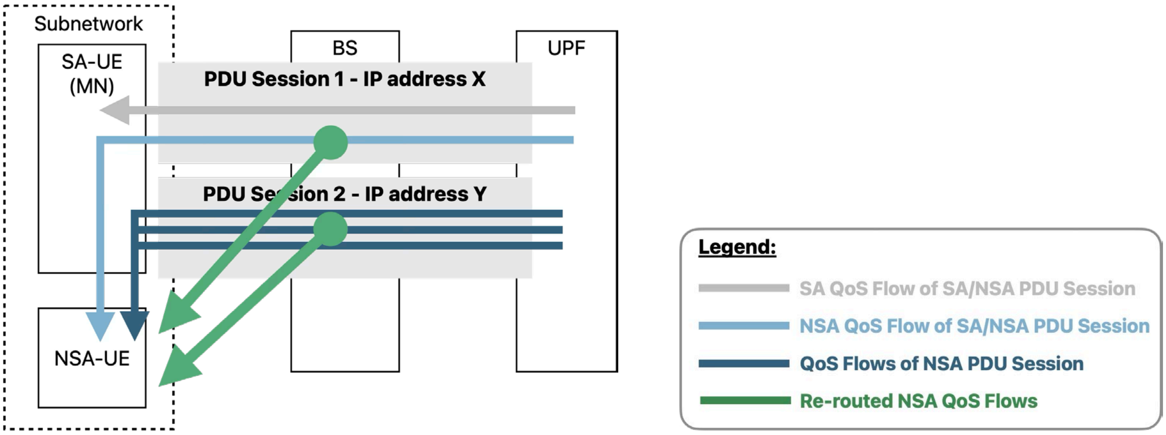

Subnetwork
SA-UE (MN)
NSA-UE
BS
UPF
PDU Session 1 - IP address X
PDU Session 2 - IP address Y
Legend:
SA QoS Flow of SA/NSA PDU Session
NSA QoS Flow of SA/NSA PDU Session
QoS Flows of NSA PDU Session
Re-routed NSA QoS Flows

Figure 10 User Plane Enhancements for NSA LC devices – DL re-routing

3.1.2.3 Security for NSA-UEs

The communication between NSA-UEs and the NW shall be as secure as communication between the NW and the regular UEs, although those NSA-UEs may have limited capabilities in terms of Control Plane functionality, e.g. in terms of supported RRC procedures or establishment and management of Access Stratum (AS) security [4]. In this subsection, a new scheme for AS Security Establishment for NSA-UEs is presented. As shown in Figure 11, it requires a new Key Derivation Function (KDF) that allows to derive corresponding keys at the SA-UE device side as well as at the BS side in order to provide a secure direct link between BS and the NSA-UE as well as between the SA-UE and NSA-UE.

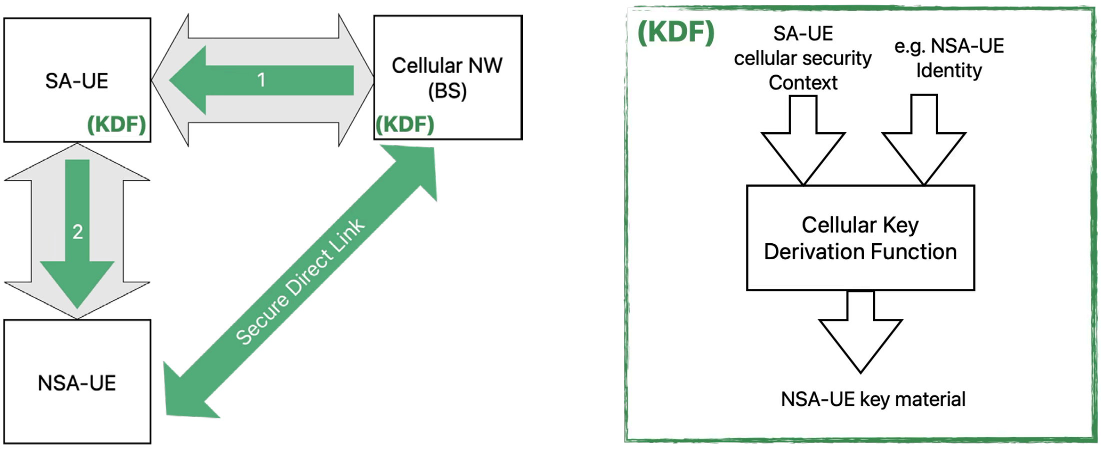


Figure 11 NSA AS Security Architecture

As described in the Section 3.1.2.1, the cellular NW provides NSA-UE configurations required for establishing the direct links to its serving NSA-UEs. This may also include a security configuration, e.g. ciphering and integrity protection algorithms highlighted as (1) in Figure 11. After this exchange, the BS and the SA-UE shall perform the new NSA-UE key generation via the proposed KDF. This could be done using the already established AS security between SA-UE and BS and additional parameters like the NSA-UE Identity. The respective keys are generated in a similar way to K_UP_enc, K_RRC_enc in 5G [5]. As a result, the same NSA-keys exist on both the BS and the SN side. The SA-UE shall provide the NSA-UE with the necessary information as annotated with (2) in Figure 11. It should be noted that this derivation function could as well be executed by the NSA-UE directly. In this case, the NSA-UE security context can be seen as a derivation of the AS context. As the NSA-UE does not have any direct interaction with the Core NW (CN), the AS context is established by the SA-UE.

3.1.3 Subnetwork Authentication

The current state of the art solutions to authenticate UEs depend on the CN support. In 5G, authentication is controlled by the CN, e.g. AMF/AUSF [5], or by dedicated servers like Proximity Services (ProSe) application servers [9]. In order to enable the new UCs [1][2], which require CN independence to achieve SN survivability in the absence of overlay 6G NW, the authentication process should be brought closer to the UEs. With this architectural enhancement, the following is achieved:

- Lower NW load: due to less NW interaction
- Less NW complexity: no need to deploy special NF/servers in the CN
- SN independence (Survivability)
- Faster authentication: due to shorter distances

If one UE (e.g., UE1) wants to act as MgtN and form a SN, and another UE (e.g., UE2) wants to join that SN, they need to trust each other and authenticate each other’s identity. Hence, they need a UE-to-UE Authentication to help establish the necessary trust while forming a SN.

Figure 12 depicts this new scheme for UE-to-UE authentication on a high level identifying three alternatives:

- Authentication via prior pairing
- Authentication with BS assistance

- Authentication via application layer.

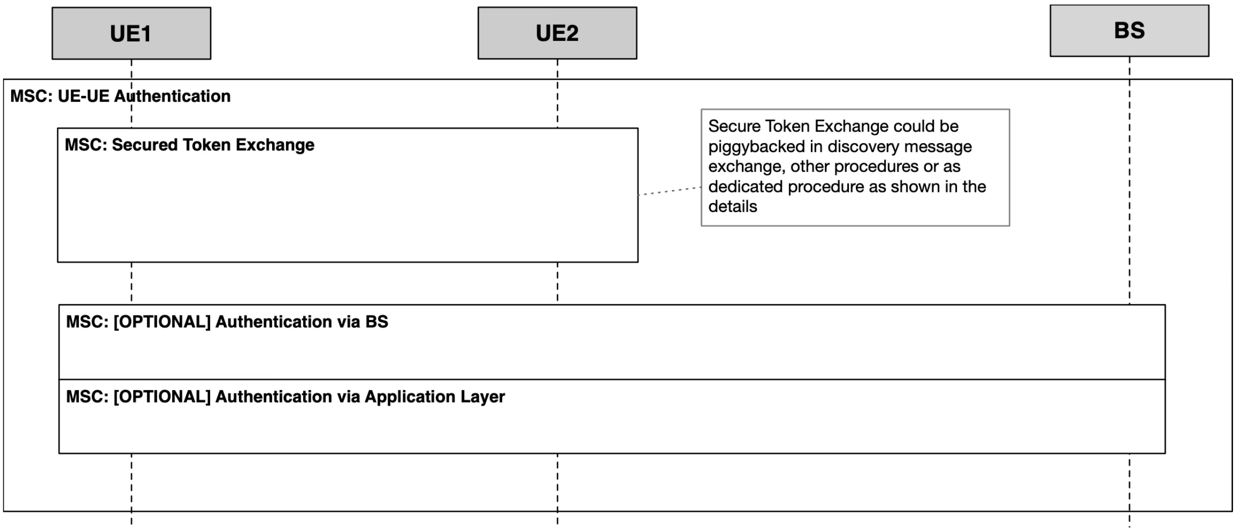

UE1
UE2
BS
MSC: UE-UE Authentication
MSC: Secured Token Exchange
Secure Token Exchange could be piggybacked in discovery message exchange, other procedures or as dedicated procedure as shown in the details
MSC: [OPTIONAL] Authentication via BS
MSC: [OPTIONAL] Authentication via Application Layer

Figure 12 UE-to-UE Authentication Schemes

It should be noted that this authentication is a general process that could either be a separate message exchange as shown in the following MSCs, or it may be incorporated as part of other procedures. For example, the authentication information could be embedded in SN discovery messages.
In the following, the UE-to-UE authentication is described as a standalone procedure.

3.1.3.1 Authentication via prior pairing

In this scheme, the UEs have pre-shared a set of keys (for example via NFC/BT) that can be used to authenticate each other. This case may not require any enhancements in 6G and, thus, it is not investigated in detail here.

3.1.3.2 Authentication with BS assistance

The different UCs described in [1][2] often assume that the involved UEs are regular 3GPP-compliant UEs that may form a SN. In this case the UEs are registered and authenticated by the NW. This could be used to establish mutual trust between two such UEs, while forming the SN. Specifically, for UEs in Connected/Inactive mode, the BS contains an AS security context. The procedure for authentication with BS assistance suggests that the BS can act as an authentication authority to enable mutual authentication between the UEs. The BS can reuse the AS keys it has established for RAN-security for each individual UE as depicted in the message sequence chart of Figure 13, where one of the UEs (e.g. the UE1 acting as MgtN) communicates with the BS in order to ensure mutual authentication.

Figure 13 Authentication with BS assistance

As already mentioned, UE1 and UE2 already have their own security contexts and a corresponding one at the BS side, i.e. the keys K_UE1_BS and K_UE2_BS in Figure 13. Still referring to the same figure, the yellow boxes indicate that UE1's security context was used while the blue ones indicate that UE2's security context was used. The procedure starts with the secure token exchange, where each of the UEs creates an authentication token using its own security context that is shared with the BS. After that, the UEs exchange those tokens with each other and store them for later verification. The actual authentication with BS assistance then happens, through one UE sending an Authentication Request towards the BS. In the example of Figure 13, this request is sent by UE1 using its secure connection with the BS. The BS shall try to verify the provided UE2 authentication token using UE2s security context, which is as well stored at the BS. After that, it shall generate an authentication token for UE2 and encrypt that using UE2s security context to avoid eavesdropping or manipulations by UE1 or by a man in the middle. The result of UE2 authentication verification and the encrypted authentication token for UE2 shall be sent back to UE1 via the Authentication Response message. In case of successful authentication verification of UE2 by the BS, UE1 shall forward the encrypted token to UE2. Finally, UE2 can conclude UE1's authentication based on the decrypted token it received.

3.1.3.3 Authentication via application layer

Similar to the scheme using the BS Assistance for authentication from the previous subsection, the UEs may establish trust by contacting a service from the cloud in the application layer. In that case, the flow is similar to Figure 13 with a third-party application server taking over the role of the BS.

3.1.4 Subnetwork IP Connectivity

UEs registered with the 6G NW usually have IP connectivity that is anchored at the UPF [18]. When such devices want to communicate with each other via their IP addresses, the traffic has to leave the RAN and go via the UPF or even through internet nodes to be routed back based on IP addresses. This happens, despite these devices being in the same cell. To enable local communication for devices in close proximity, 6G shall enable more direct communication, where traffic gets re-routed locally. An example is shown in Figure 14, where (re-)routing can happen at the UE for a D2D link (1), at the MgtN for traffic within SN (2) or at the BS (3). This enables low latency communication among devices in a SN as described in D2.2 [2]. The UEs in a SN may already have local links established; however, new mechanisms for seamless switching between local links and global communication should be defined, especially when considering devices enter or leave SNs dynamically.

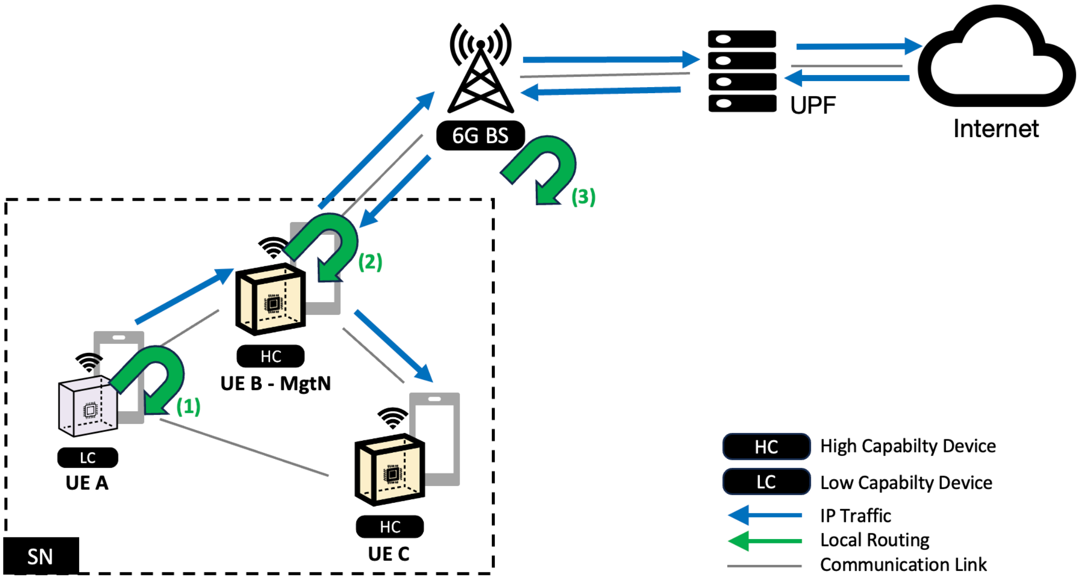


Figure 14 IP traffic routing via UPF/internet and local re-routing.

This approach focuses on UEs having a regular internet connection. The UEs get an IP address assigned during the registration process when the PDU Session is established [18] and the communication is based on these IP addresses. As a side note, UEs may either get a global IP address, or a local IP address that gets mapped into a global IP address using Network Address Translation (NAT) at the UPF. In the latter, a new control plane procedure is required for UEs requesting their global IP addresses. To preserve user privacy, the IP address information required for local re-routing of traffic shall only be shared among the UEs that communicate with each other, not the MgtN nor the BS. Consequently, there is a need for mechanisms enabling the UEs to request and configure local routing for certain parts of their outgoing traffic, e.g., towards the MgtN, without revealing IP address information.

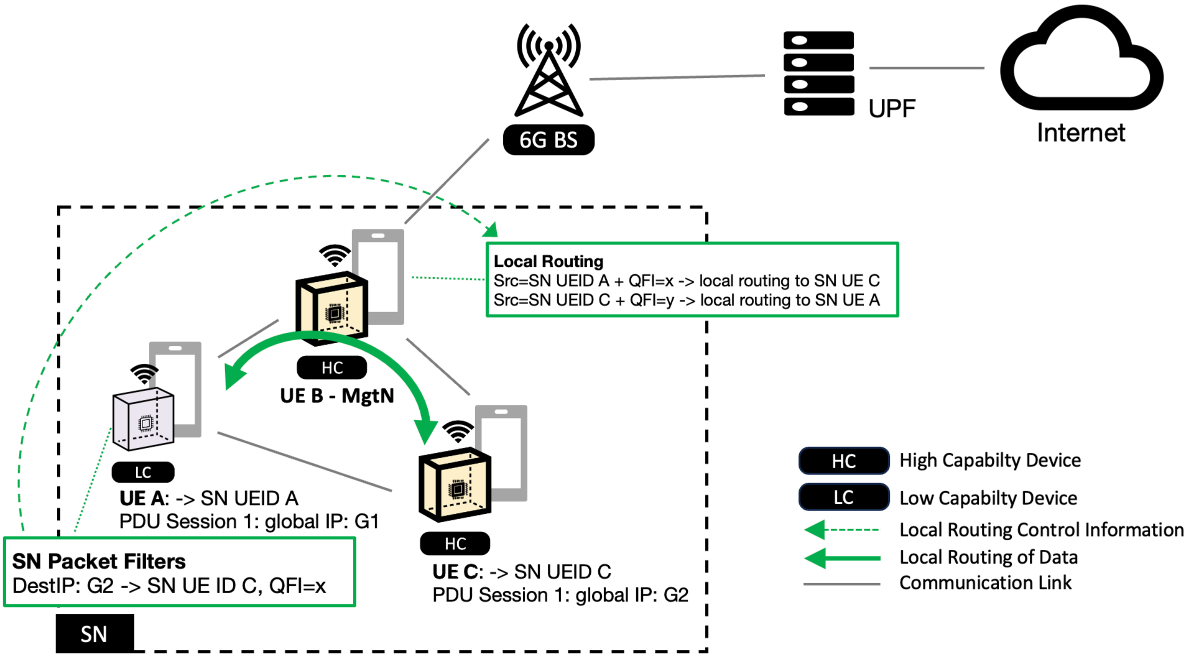

6G BS
UPF
Internet
Local Routing
Src=SN UEID A + QFI=x -> local routing to SN UE C
Src=SN UEID C + QFI=y -> local routing to SN UE A
HC
UE B - MgtN
LC
UE A: -> SN UEID A
PDU Session 1: global IP: G1
HC
UE C: -> SN UEID C
PDU Session 1: global IP: G2
SN Packet Filters
DestIP: G2 -> SN UE ID C, QFI=x
SN
HC High Capabilty Device
LC Low Capabilty Device
Local Routing Control Information
Local Routing of Data
Communication Link

Figure 15 Local routing with NAT at the UPF

As a first step, global IP addresses shall be shared only among the involved SN UEs via a new control plane procedure that could be part of *Subnetwork Control Plane* (snCP) as suggested in D4.2 [4]. As a second step, UEs, which have identified that some communication can be routed locally, shall set up new traffic filters to distinguish such packets and label them accordingly with new “Local QFIs” (e.g. in SDAP [17] or the proposed SN-RP discussed in [4]). Note that SDAP protocol enhancements are required to distinguish “NW configured QFIs” from “Local QFIs” to avoid collisions. Finally, the mapping of Local QFI and the targeted SN UE ID can be shared with the MgtN to allow re-routing based solely on QFI labels without revealing any IP address information or requiring IP routing functionality. In case NAT is active at the UPF, it may require additional local address translation at the sending UE. The required functionalities in the protocol stack of a sending UE, and the MgtN as anchor for the local re-routing are shown in Figure 16, where UE A is portrayed as the sending UE.

Figure 16 Local Routing functionality in the protocol stack

When a UE leaves the SN, local re-routing by the MgtN ends and the packets can continue via the UPF as highlighted in Figure 14. This scheme of setting up local re-routing at the MgtN can also be applied to the BS as shown in see Figure 14 (3), where the SN UE IDs and local QFIs can be shared with the BS enabling the local re-routing at RAN level.

3.1.5 Summary

In this section enhancements in the SN architecture have been proposed, which enhance the UE autonomy and independence from the parent network, while at the same time guaranteeing seamless integration into the parent network. In this context, the concept of NSA UEs has been enabled with three procedural enhancements, namely those of NSA-UE configuration, PDU sessions multiplexing as well as a framework with the novel KDF mechanism for secure end-to-end communication between the parent NW and the NSA-UE.

UE autonomy and independence have been further enhanced with the introduction of the novel device-to-device authentication framework. Three modes have proposed: authentication with prior pairing, via the BS SA security context or via security context provided at the application layer. On all three modes there is no interaction with the CN, thus making the overall authentication process faster, while decreasing NAS overhead signalling.

Finally, architectural enhancements, enabling local IP routing, have been proposed by introducing SN IP addresses as well as local QFIs. These SN IP addresses along with the local QFIs are utilized in the SN-RP protocol of [4] so that local data stays exclusively within the SN without any parent network involvement. Therefore, the UE privacy and autonomy are enhanced, while achieving at the same time a reduction in latency.

3.2 SUBNETWORK FORMATION AND MOBILITY

3.2.1 Introduction

The architectural enhancements introduced int the previous section and the mobility procedures introduced in [4] have not addressed a crucial issue, that of how an SN is formed in the absence of MgtNs. This issue is addressed in Section 3.2.2, where a decentralized procedure on how nodes agree which will become an MgtN, which in turn form a subnetwork. Subsequently in Section 3.2.3, the specific procedures are introduced, which enable a UE to select the appropriate MgtN and by extension the appropriate subnetwork. A final enhancement is also proposed in Section 3.2.4, where the SN assists in the mobility procedures of a UE in the parent 6G NW.

3.2.2 Subnetwork Formation

3.2.2.1 UE-centric Subnetwork Formation

So far, the underlying assumption has been made that there are eligible nodes that have assumed the MgtN role. In fact, in Sidelink (SL), this decision is made by the NW or the Proximity Services (ProSe) [10] server. However, this dependence from the NW may not be suitable for the dynamic use cases in focus [2], where independence from the NW is required to ensure the SN's survivability. To address this, a user-centric approach should be followed, where the BS and the 6G NW may not be involved. Hence, there is an absence of a central authority, leading to an inherently decentralized solution. Consequently, the problem statement becomes as follows: given several UEs of various capabilities in terms of power, computation and services, how to select the MgtNs to optimize the individual UE's performance.

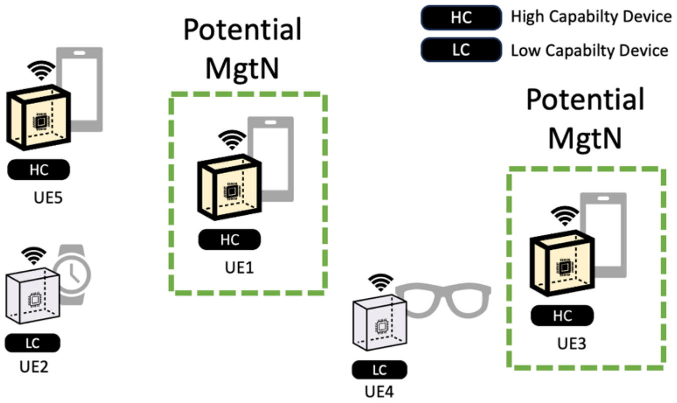


Figure 17 User-centric subnetwork formation topology.

An exemplified topology is shown in Figure 17, where UEs of different capabilities are shown. UE1 and UE3 are adequately capable to support the MgtN role. By contrast, UE5, albeit a HC UE, does not have the intention to become a MgtN. It should be noted that the intention whether to become a MgtN is based on the UE's internal function and internal state, e.g., its power levels and its computational capabilities.

To achieve a decentralized scheme for MgtN selection, negotiations have to take place among the UEs with intent to become MgtN, so that they collectively decide which nodes will indeed act as MgtN. The message sequence chart for the decentralised SN formation is shown in Figure 18. Note that it corresponds to the exemplified topology shown in Figure 17. For the sake of simplicity, only UE1, UE2 and UE3 nodes are shown in Figure 18. Explicitly, UE4 and UE5 will act exactly as the LC UE2. Additionally, UE2 is assumed not to be directly reachable by UE3.

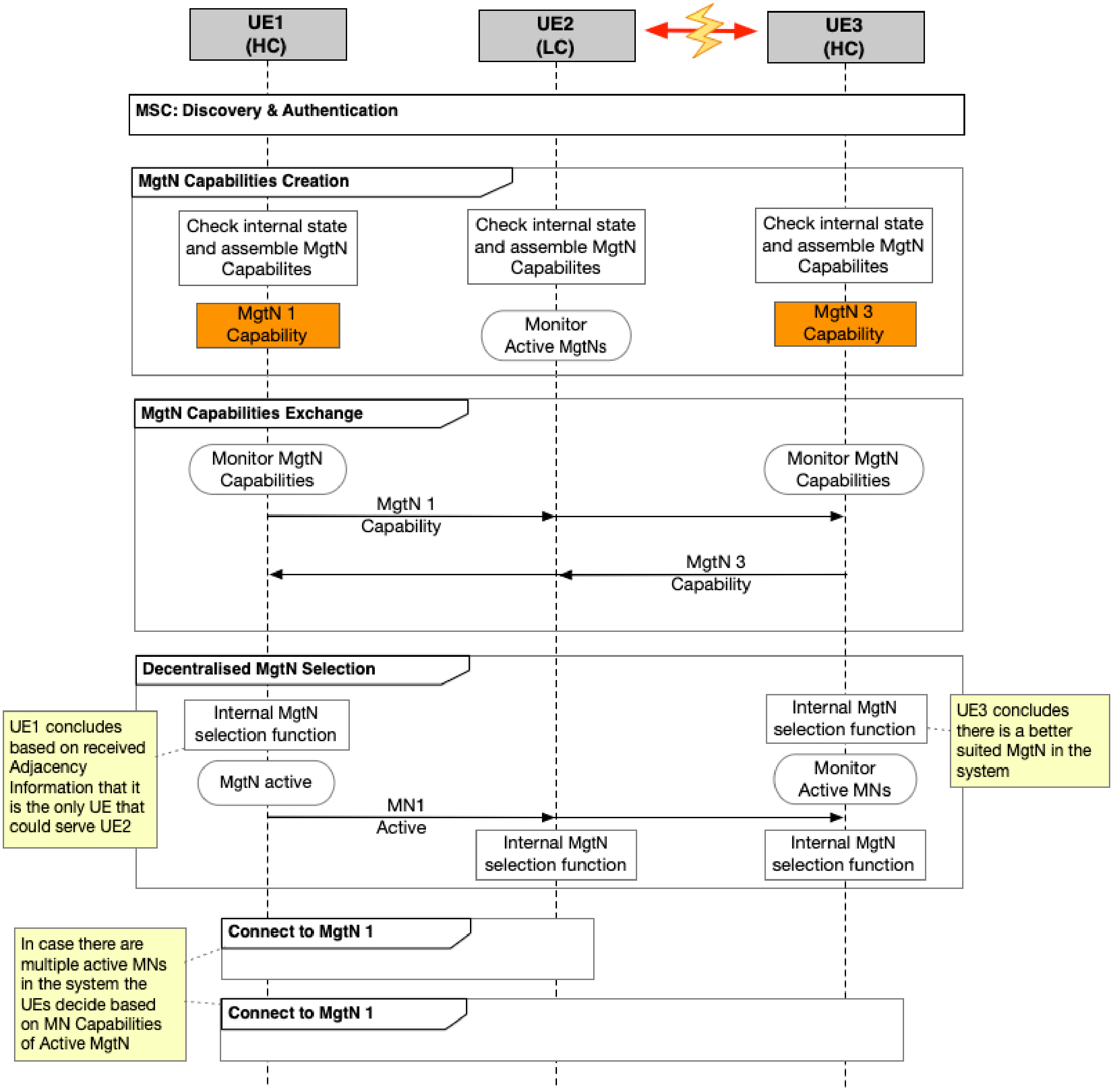

UE1 (HC)
UE2 (LC)
UE3 (HC)
MSC: Discovery & Authentication
MgtN Capabilities Creation
Check internal state and assemble MgtN Capabilites
Check internal state and assemble MgtN Capabilites
Check internal state and assemble MgtN Capabilites
MgtN 1 Capability
Monitor Active MgtNs
MgtN 3 Capability
MgtN Capabilities Exchange
Monitor MgtN Capabilities
Monitor MgtN Capabilities
MgtN 1 Capability
MgtN 3 Capability
Decentralised MgtN Selection
UE1 concludes based on received Adjacency Information that it is the only UE that could serve UE2
Internal MgtN selection function
Internal MgtN selection function
UE3 concludes there is a better suited MgtN in the system
MgtN active
Monitor Active MNs
MN1 Active
Internal MgtN selection function
Internal MgtN selection function
In case there are multiple active MNs in the system the UEs decide based on MN Capabilities of Active MgtN
Connect to MgtN 1
Connect to MgtN 1

Figure 18 Message sequence chart of decentralised subnetwork formation.

At first, a discovery and authentication phase take place as presented in Section 3.1.3, where the UEs discover their neighbouring UEs and authenticate each other's identities. Moving on to the "MgtN Capabilities Creation" block in Figure 18, the UEs determine whether they have an intention to become MgtNs. As already mentioned, this intention whether to become a MgtN is based on the UE's internal function and internal state. In the example of Figure 18, UE3, being an LC UE, determines that it will not become a MgtN and begins to monitor for reference signals by active MgtNs. By contrast, UE1 and UE2 determine that they can become MgtNs and assemble their respective MgtN capabilities. The latter are a set of each UE's capabilities for supporting and operating the SN. These capabilities are encapsulated in the so-called "MgtN Capability Report", which contains:

- The respective temporary UE ID, as identified in the authentication process.
- Adjacency information: this element is a list of the UE's neighbours' temporary IDs without revealing their identity. This list is subject to the privacy constraints, i.e., UEs may not consent in sharing their neighbourhood relation to or beyond their direct neighbours. The list may include information about e.g., the neighbours' authentication status or their link RSRP value.
- Communication Capabilities:
    - Connection to NW, e.g. LTE, NR, 6G or NTN

  - Number of supported Carrier Components (CC)
  - Round Trip Time (RTT) to BS
  - Mobility State.
- Computational Capabilities for SN Use
  - Memory Size Allocation
  - Floating Point Operations Per Second (FLOPs)
  - Battery Energy.
- Functional Offloading Services: a list for functional offloading applications to/being? offered upon SN formation.

In the “MgtN Capabilities Exchange” phase of Figure 18, each UE volunteering to become an MgtN broadcasts their MgtN Capability Report, thus notifying their neighbouring UEs of their intent to become an MgtN. During this phase, all potential MgtNs collect MgtN Capability Reports from neighbouring HC UEs volunteering to become MgtNs. Subsequently, the “Decentralised MgtN Selection” process takes place, as seen in Figure 18. During this phase, UE1 and UE3, upon receiving a set of MgtN Capability Reports, use their internal function to determine whether they will become active MgtNs. Explicitly, this internal function takes as inputs the UE’s received set of MgtN Capability Reports as well as their set of applications and requirements, derived by the UE’s internal state. The internal function invokes a model determining whether the UE will become an active UE. This model can be deterministic, heuristic or even an AI/ML approach, such as a deep learning method. Should the UE decide to become an active MgtN, as in UE1’s case, it begins to transmit MgtN-related reference signals, notifying its neighbouring UEs that they can connect to it. Otherwise as in UE3’s case in Figure 18, the UE begins monitoring for MgtN-related reference signals to connect to a MgtN. Finally, the UE connection process into the SN takes place, which is elaborated in Section 3.2.3, where the UEs monitoring for MgtN-related reference signals select the best MgtN to connect. Note that UE3 can decide to become an MgtN based on its own internal function. In this case, the UEs would have to select between the SNs of UE1 and UE3.

Note that no NW involvement has taken place throughout this decentralised formation process, thus providing the SN architecture of [3] with NW independence, therefore achieving survivability in the absence of an overlay network.

3.2.2.2 RAN-supported Subnetwork Formation

SNs are entities formed by devices individually on a voluntary basis and ideally between devices directly in a D2D fashion. These devices would need to have additional functionality: they should be capable of searching and discovering nearby devices. However, this may also come at the expense of increased power consumption. In this subsection, a new support function is described that shall be provided by 6G BS or within the RAN. This support function helps UEs within a certain area to find and discover each other, enabling them to form a SN in a more power efficient way, while at the same time preserving user privacy. In the proposed scheme, depicted in Figure 19, the 6G BS provides a new SN Formation function and acts as an independent information broker, that aims to connect trusted users without exercising any control.

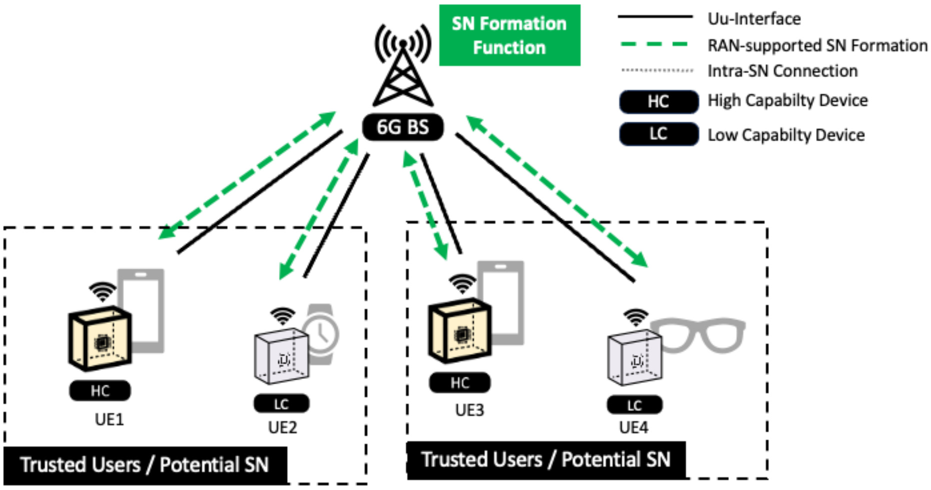


Figure 19 RAN supporting devices in SN formation

The communication flow of this anonymized BS-aided SN Formation is shown Figure 20. It is assumed that UEs forming a SN have already established some sort of trust among each other and have exchanged some information, e.g. some anonymized IDs, among each other. These UEs are referred to as trusted users in the context of this subsection. All UEs that aim to form a SN shall inform their BS about their intention to form a SN. Therefore, they shall provide an ID known to their trusted users, potentially an MgtN Capability as well as a list of IDs belonging to their trusted users. The BS may receive multiple search requests from different UEs and shall maintain a list of "searching UEs" and their "trusted Users" by storing their temporary IDs. In addition, the BS may store the provided MgtN capabilities for later distribution.

If there is a match of "searching UEs" and "trusted Users", the BS shall inform trusted users and distribute the respective MgtN Capabilities among them in order to support the SN Formation and MgtN selection process. It is up to the UEs to select the MgtN based on the received MgtN Capability of their trusted users and start the respective SN registration. This is further elaborated in Section 3.2.2.3.

Figure 20 Message sequence chart for RAN-supported subnetworks formation

3.2.2.3 Subnetwork Registration

In principle, a SN can exist independently from the 6G NW, such as when there is no coverage. In addition, a SN may register itself at the 6G BS to qualify for communication resource granted from the 6G NW, as elaborated in Section 5.1. The SN registration procedure provides this integration. Explicitly, there are two different variants of SN registration, a MgtN-centric and a RAN-assisted approach. In the MgtN-centric solution, the MgtN informs the 6G NW about the SN and its serving devices, as shown in Figure 21. The 6G BS provides in return the SN with potential communication resources, which are distributed within the SN by the MgtN. During this process, the 6G BS also assigns SN UEIDs for the Virtual Connections and UE Contexts as they have been proposed in Section 2.2.1.3 of D4.2 [4].

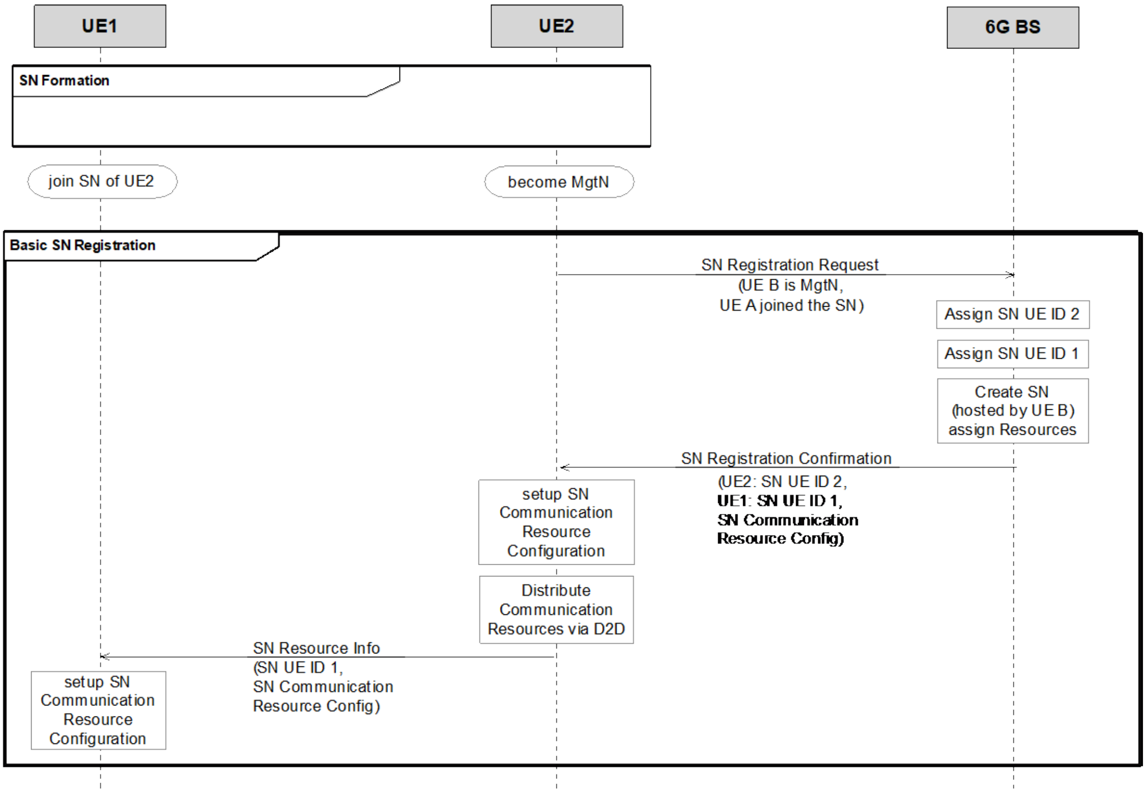


Figure 21 Message sequence chart for MgtN-centric SN registration

As far as the RAN-assisted solution is concerned, it is presented in Figure 22, where the MgtN registers only itself with the 6G BS. The MgtN requests the BS to distribute the SN communication resources to MgtN's trusted users along with the SN UE ID assignment of Section 2.2.1.3 of D4.2 [4]. The 6G BS then sends directly a message to the to MgtN's trusted users, i.e. UE1 in Figure 22, including the recipient's SN ID as well as the SN resource configuration. In both cases, after successful SN registration, the SN can operate on granted SN communication resources, as highlighted in section 5.1.

Figure 22 Message sequence chart for RAN-supported SN registration

3.2.3 Subnetwork Selection

In terms of mobility, the interactions of the UE with the MgtN and the parent network to maintain the so-called virtual connections have been presented in D4.2 [4]. Nevertheless, the processes and criteria, used by the UE to select which SN to join, have not been covered in D4.2 [4]. Based on the architectural considerations of D2.4 [3], one of the most important goals in the SN design is the increase in the UE autonomy. With this consideration in mind, a user-centric framework for SN selection without any parent network involvement is presented in this section. This is in direct contrast to the methods used in Sidelink (SL) Relay [10], where the processes of relay selection as well as that of establishing links is under tight network control. This dependency from the parent network creates significant overheads in dynamic topologies such as the consumer use case [2], where nodes have a degree of mobility. Under the current framework, the BS of the parent network should configure each of the UEs with measurement configurations for SL relays in the vicinity of each UE. The UEs shall perform these measurements and then forward the associated reports to the BS. In this case, the BS acts as a central authority for mobility decisions. Therefore, not only is delay imposed due to the propagation of these reports to the BS but also scalability issues arise, especially in dense deployments.

Additionally, SL relay selection reports include measurement quantities from NR [11], such as time-or frequency-domain Reference Signal Received Power (RSRP), Reference Signal Received Quality (RSRQ) or Signal to Interference plus Noise Ratio (SINR). These metrics and the current reporting framework in 3GPP are centred around link quality and fail to address additional SN-related functional and computational offloading capabilities, which were outlined in [2],[4]. Consequently, the current framework may be deemed as insufficient and further enhancements are required.

In this section, a more user-centric mechanism for SN selection is introduced, where the selection decision lies at the UE side. Furthermore, a new flexible mechanism for SN capability exchange is introduced to perform a better selection based on the UE's specific needs and requirements. Finally, the inter-SN mobility is enhanced through a mechanism of capability exchange across SNs.

3.2.3.1 User-centric Subnetwork Selection

Before delving into the specifics of the new SN selection scheme, it is worth mentioning the assumptions made for this novel selection process. A single UE has the intention to join a SN. In the UE's vicinity there exist several devices acting as MgtNs and managing their respective SNs.

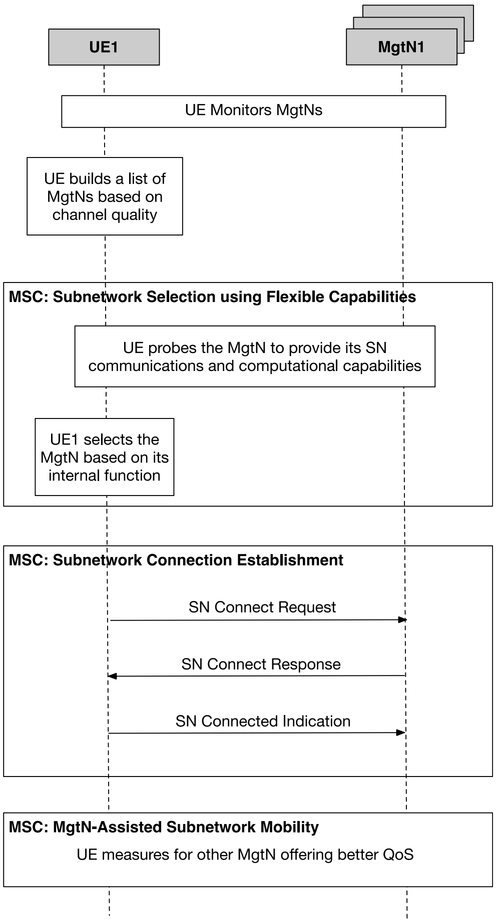

UE1
MgtN1
UE Monitors MgtNs
UE builds a list of MgtNs based on channel quality
MSC: Subnetwork Selection using Flexible Capabilities
UE probes the MgtN to provide its SN communications and computational capabilities
UE1 selects the MgtN based on its internal function
MSC: Subnetwork Connection Establishment
SN Connect Request
SN Connect Response
SN Connected Indication
MSC: MgtN-Assisted Subnetwork Mobility
UE measures for other MgtN offering better QoS

Figure 23 User-centric subnetwork selection utilizing the subnetwork capabilities report transmitted by MgtNs.

An overview of the proposed SN selection mechanism is shown in Figure 23. At first, the UE monitor for neighbouring MgtNs. This process involves monitoring for broadcast signalling from MgtN, such as *Synchronization Signal Blocks* (SSB), *Master Information Blocks* (MIB) and System Information Blocks (SIB), or even dedicated messages, such as discovery messaging between the MgtNs and the UE. For the latter, the legacy Proximity Services (ProSE) [9] could be leveraged. Based on this signalling, the UE builds and maintains a list of eligible MgtNs based on signal quality in terms of e.g., RSRP, SINR or RSRQ. The rationale behind creating this list is to limit the complexity by excluding MgtNs that do not provide adequate link quality.

After the creation of the eligible MgtNs list, the UE probes each of the MgtNs to provide the UE with their set of SN capabilities. Each of the MgtNs then sends back to the UE a payload containing their SN capabilities both in terms of communication as well as of computation:

- **Subnetwork Communication Capabilities, e.g.**:
    - Connection to parent network
    - Connection quality to the overlay network quantified in RTT classes
    - Subnetwork load
    - Number of CCs supported
- **Subnetwork Computation Capabilities, e.g.:**
    - Computational Resources
    - Memory Resources
    - Computational Precision

The UE uses those capability reports, the channel quality of the MgtNs as well as its own communication and computation requirements as inputs into its internal function to select the optimal MgtN, as seen in Figure 23. An example of such requirements could be as follows:

- **Communication requirements, e.g.:**
    - Traffic type
    - Minimum bit rate
    - Minimum latency
- **Computation requirements, e.g.:**
    - Minimum complexity
    - Minimum memory
    - Minimum latency
    - Minimum Precision

Note that these requirements are derived from all the UE's active applications. As far the UE's internal function is concerned, it is portrayed in Figure 24. There, the aggregated capability reports from all eligible MgtNs along with the UE's communication and computation requirements are input into an evaluation model $\mathbb{M}$. The latter is UE-vendor-specific and could be:

- a deterministic model, e.g. a water-filling method
- a heuristic model, e.g. a stochastic gradient-descend-based algorithm
- a neural network, e.g. a Deep Neural Network (DNN)

Additionally, this model could be either trained offline or calibrated with online updates. The output of the model is a soft evaluation vector, with each of its elements corresponding to the utility of a specific MgtN $\mathrm{a}_{\mathrm{MgtNx}} \in (0,1)$. Subsequently, the optimal MgtN with the maximum utility is selected.

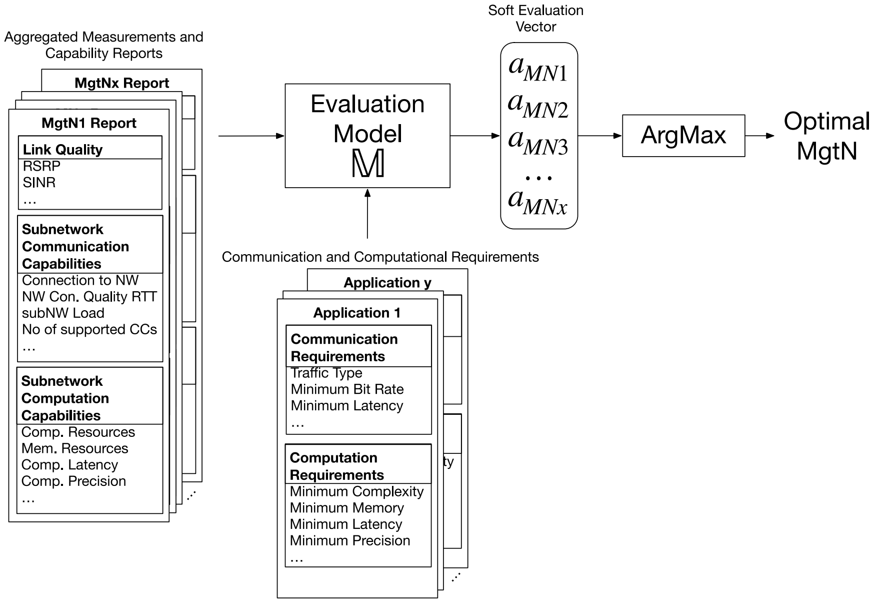


Figure 24 UE internal function for selecting the optimal MgtN.

Once the optimal MgtN is selected, the UE sends a *SN Connection Request* to the chosen MgtN, where the UE includes UE specific requirements on certain MgtN capabilities, as seen in Figure 23. Should the MgtN accept the UE into its SN, it sends back to the UE a *SN Connection Response* with the SN configuration. Upon proper configuration, the UE replies to the MgtN with *a SN Connected Indication*, which implies that the UE is connected to the SN managed by the MgtN. In case the UE is not admitted by the MgtN, it will keep sending *SN Connection Request* to the next optimal MgtNs until it connects to a SN. Upon connection, the UE may keep measuring for other MgtNs in search of better Quality of Service (QoS) compared to its serving MgtN.

Naturally, this process works without any parent network involvement, thus minimizing the associated delays and keeping control at the UE side. Additionally, the selection process is enhanced by the transmission of the SN capabilities to the UEs.

3.2.3.2 MgtN-assisted subnetwork mobility

One potential issue with the selection process, which was presented in the previous subsection, may be the delay associated with gathering the SN capability reports from all the neighbouring MgtNs. In fact, the UE would have to probe each of the MgtNs individually to gather these reports. Apart from increased delay, this results in excessive power consumption due to repetitive signalling.

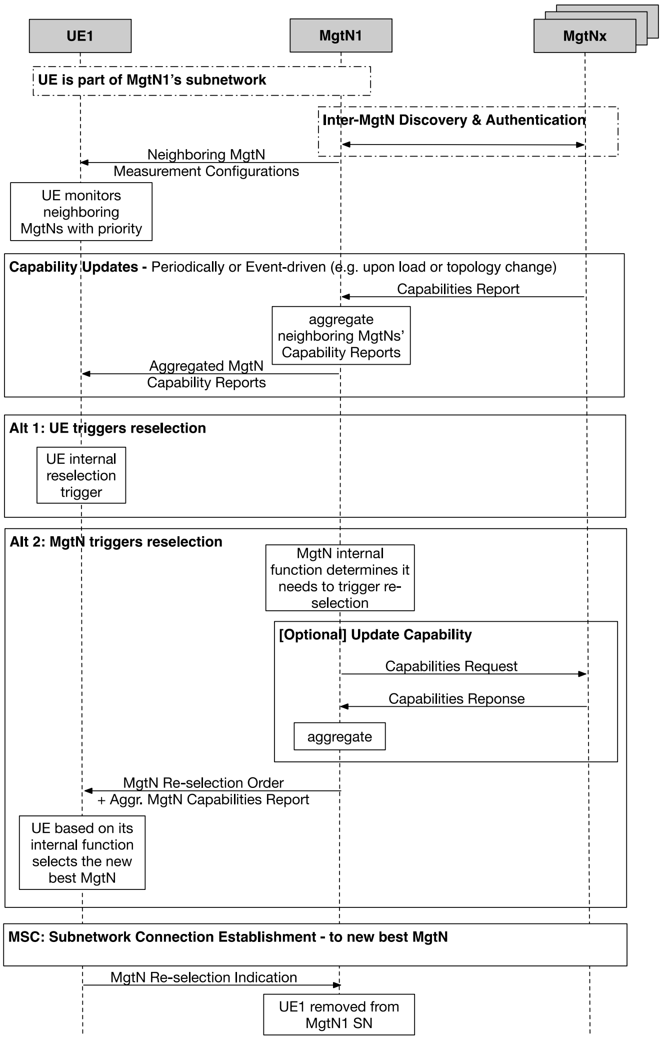

UE1
MgtN1
MgtNx
UE is part of MgtN1's subnetwork
Inter-MgtN Discovery & Authentication
Neighboring MgtN Measurement Configurations
UE monitors neighboring MgtNs with priority
Capability Updates - Periodically or Event-driven (e.g. upon load or topology change)
Capabilities Report
aggregate neighboring MgtNs' Capability Reports
Aggregated MgtN Capability Reports
Alt 1: UE triggers reselection
UE internal reselection trigger
Alt 2: MgtN triggers reselection
MgtN internal function determines it needs to trigger re-selection
[Optional] Update Capability
Capabilities Request
Capabilities Reponse
aggregate
MgtN Re-selection Order + Aggr. MgtN Capabilities Report
UE based on its internal function selects the new best MgtN
MSC: Subnetwork Connection Establishment - to new best MgtN
MgtN Re-selection Indication
UE1 removed from MgtN1 SN

Figure 25 MgtN-assisted subnetwork reselection.

The proposed solution to this issue is to offload the functionality of gathering the MgtN capabilities to the serving MgtN after joining a SN. This process, referred to as *MgtN-assisted SN reselection*, is portrayed in Figure 25. As presented in D4.2 [4], the MgtNs can discover and establish links with neighbouring MgtNs. They can also authenticate with each other using the mutual authentication process discussed in Section 3.1.3. After the UE joins the SN, the serving MgtN provides the UE with refined measurement configurations for neighbouring MgtNs. This enables the UE to measure reference signal by the neighbouring MgtNs and thus assess their link quality. Based on this link quality, the UE filters eligible neighbouring MgtNs. The MgtNs, having established these inter-MgtN connections, can receive the SN capability reports from their neighbouring MgtNs. These are aggregated at each MgtN and then are transmitted to the UEs of the SN. This capability update could be either periodic or event-driven due to a state change in a neighbouring SN.

After the transmission of the aggregated SN capability reports, there are two alternatives, presented as *Alt 1* and *Alt 2* in Figure 25. In *Alt 1*, the UEs use their internal function presented in Figure 24 to select an optimal MgtN. By contrast, in *Alt 2* the reselection process is triggered by the MgtN. In this case, the MgtN has its own internal function, where it decides whether to retain the role of the MgtN or not. Explicitly, this function is similar to that of Figure 24. Factors influencing the MgtN's decision could be, e.g., the MgtN's power levels or the amount of computation and communication resources for managing the SN. Upon deciding to step down as a MgtN, the MgtN sends to its serving UEs a *Re-selection Order*, as seen in Figure 25. It may optionally collect updated capabilities of the neighbouring MgtN's or re-use already gathered information and propagate these aggregated capabilities reports with a *Re-selection Order*. The UEs in turn use their internal function to select a different MgtN. Once the new serving MgtN is selected, the *Subnetwork Connection Establishment* process of Figure 23 is followed by the UEs. After successfully connecting to the target MgtN, a UE sends a *MgtN Re-selection Indication* message to the source MgtN, completing its departure from the previous SN.

Note that the MgtN can optionally aid the UE to authenticate with the target MgtN before UE starts the *Subnetwork Connection Establishment* process, using the process of Figure 13, where source MgtN plays the role of the BS, as shown in Section 3.1.3.2. An alternative to this MgtN-assisted process is to perform the process of Figure 23 as a fallback. This can happen when the UE enters a Radio Link Failure (RLF) mode due to a sudden connection loss with the MgtN.

3.2.4 User-centric predictive mobility with subnetwork support

Baseline handover (BHO) was introduced in Rel. 15 of NR [19] and allows the network to control handover decisions based on the measurement results of the UE. However, under BHO the performance depends on timing of the measurement report transmission and/or HO command reception. To address this issue, Conditional Handover (CHO) was introduced in Rel. 16 [20] with the aim of reducing interruption time. This is achieved by configuring UEs with a condition to autonomously execute the handover, hence increasing the robustness compared to BHO.

Despite the improvement achieved by CHO, there are still some issues with its current implementation in 3GPP. To begin with, the UE is configured with the RRC reconfigurations of the potential target cells earlier, i.e. before UE approaches the cell edge. The handover execution decision relies on signal measurements based on pre-configured events, such as the A3 event triggered when the target cell is stronger than the serving cell. Although the target signal power may be high, it does not guarantee that the QoS requirements of the UE will be satisfied, in case the load of the target cell is high. In this case, should target QoS requirements turn out not to be satisfied in the target cell, the UE may need to be handed over to another cell that satisfies the QoS requirements. This, in turn, results in more HOs leading to higher interruption time. Besides the QoS requirements aspect, congested cell can also lead to collisions in the random-access procedure, impacting the interruption time.

In [21], a "consecutive CHO" mechanism was proposed to enable the UE to keep the prepared cells configuration even after CHO execution. In [22], the considered objectives include keeping conditional

Primary and Secondary (PSCell) change/activation (CPC/CPA) after handover to allow subsequent cell group to change without reconfiguration of CPC/CPA. However, this creates a greater time difference between the CHO preparation and the handover execution, thus increasing the probability of load change at the prepared cells. Additionally, cell traffic data may not be possible to be propagated to the UE, since the trigger of the CHO is the UE detecting low serving cell signal quality. At that time, it is impossible for the serving cell to provide this information owing to the lack of reliable communication link.

The SNs due to their locality could still enable a reliable communication link, thus assisting in providing information for accurate cell selection, therefore improving 3GPP CHO and reducing interruption time. Moreover, the MgtNs could receive information by their serving UEs regarding their connection with the overlay 6G NW. Examples of this information include their serving cell ID, their channel conditions, their QoS indication and their mean scheduling delay. The MgtNs could then aggregate this information and share it with their neighbouring MgtNs. An exemplified topology can be seen in Figure 26, where three SNs are established, each served by a different 6G BS. In this example, UE1 is moving towards the intersection cell edges of BS2 and BS3. The MgtNs have established the D2D links annotated with the brown arrows with their neighbouring MgtNs to share their aggregated information for the overlay NW cells. In a nutshell, when the UE1 detects its serving BS1 cell quality falls below a threshold, it requests side information from its serving MgtN. MgtN1 provides this information thus enabling UE1 to make a more accurate mobility decision, thus reducing its interruption time.

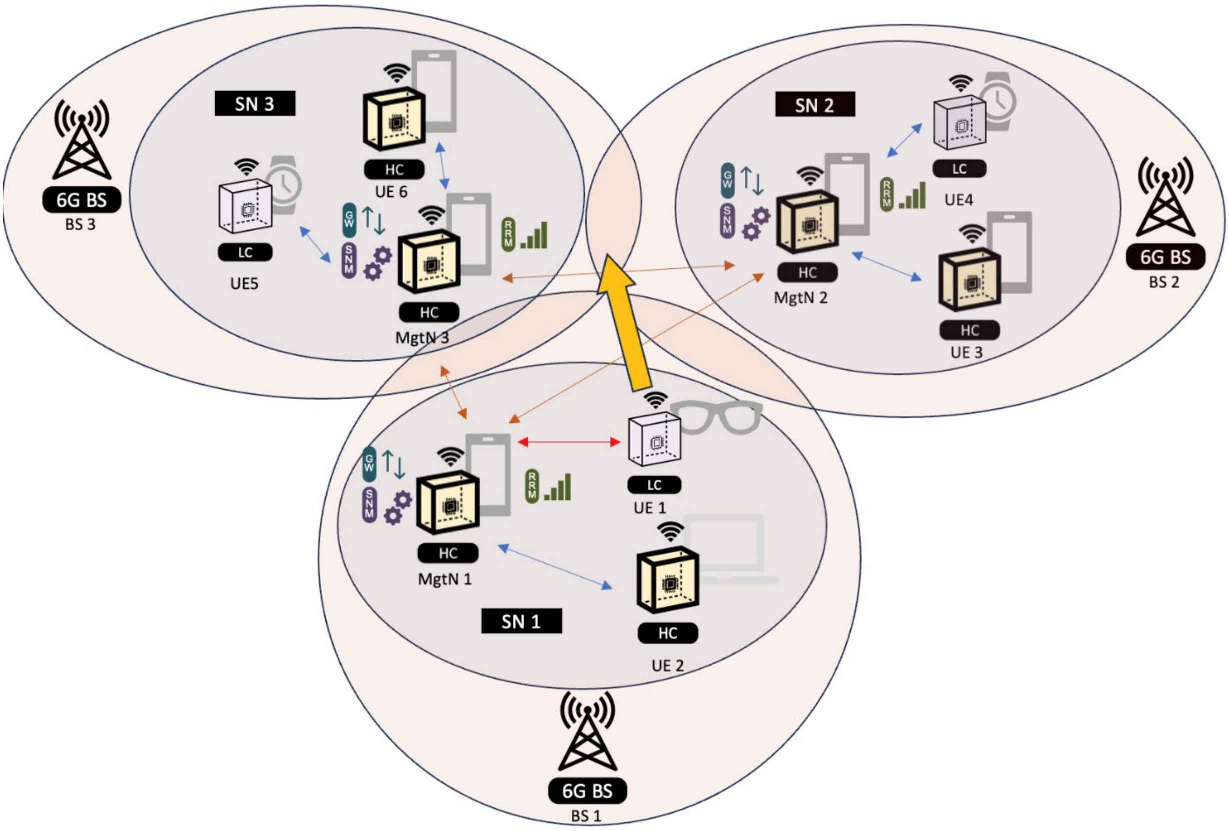


Figure 26 Subnetwork-assisted predictive mobility – An exemplified topology.

In the next subsections, detailed descriptions are provided regarding the method for performing data collection and exchange within a SN as well as the overall process of sharing this information across SNs and with the requesting UEs.

3.2.4.1 Data Collection and Exchange between MgtN and UEs

Before delving into the specifics of the SN-assisted predictive mobility signalling, the methods for enabling data collection and their exchange between UEs and their serving MgtN will be presented. The respective message sequence chart is shown in Figure 27. After the SN setup and establishment, the MgtN configures all the UEs belonging to the SN with all the relevant Quality of Experience (QoE) Information Elements (IE) to be shared in Step 1 of Figure 27. These include, among others, their QoS Indicator value [18] cell ID, TCI state, channel conditions, mean scheduling delay, DL/UL load information. Note that the UEs can be configured to report this information in a periodic or event-driven manner. In Step 2, the UEs indicate to the MgtN that the configuration has been applied thus confirming data sharing with their serving MgtN.

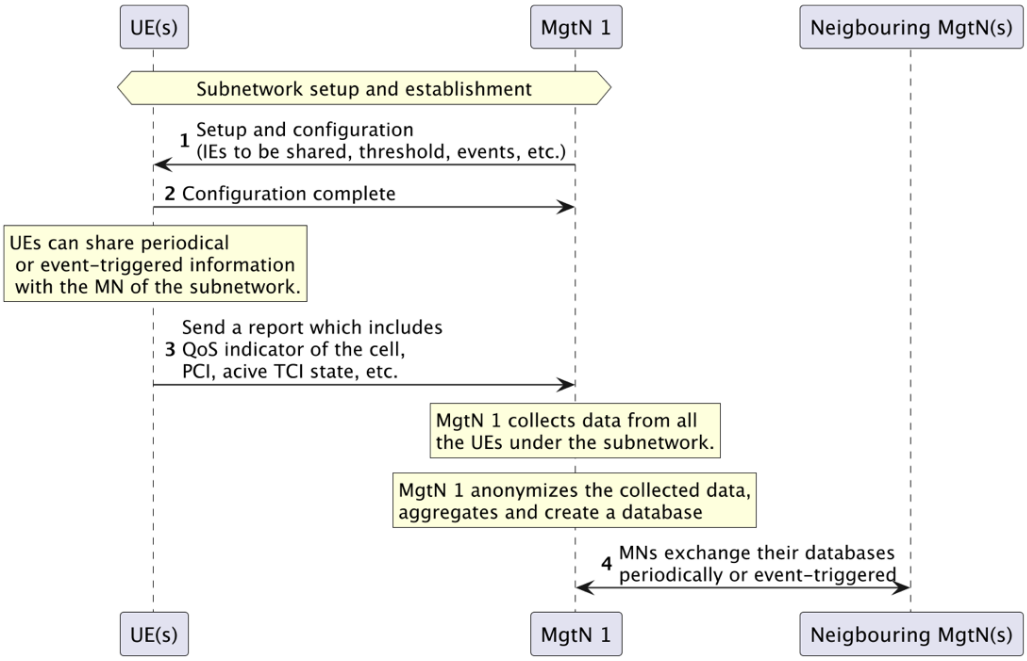

UE(s)
MgtN 1
Neigbouring MgtN(s)
Subnetwork setup and establishment
1 Setup and configuration
(IEs to be shared, threshold, events, etc.)
2 Configuration complete
UEs can share periodical
or event-triggered information
with the MN of the subnetwork.
Send a report which includes
3 QoS indicator of the cell,
PCI, acive TCI state, etc.
MgtN 1 collects data from all
the UEs under the subnetwork.
MgtN 1 anonymizes the collected data,
aggregates and create a database
4 MNs exchange their databases
periodically or event-triggered
UE(s)
MgtN 1
Neigbouring MgtN(s)

Figure 27 Message sequence chart of the data collection and exchange between UEs and their serving MgtN

Upon successful configuration, the UEs start collecting the configured QoE IEs and transmit the respective reports to the MgtN in Step 3. A potential instantiation of the IE comprising the QoE report is as follows:

```
QoEDevice ::=                       SEQUENCE {
                devId               INTEGER (0 ..255),
                PhysCellId          INTEGER (0 ..1007),
                TCI-StateId         INTEGER (0..maxNrofTCI-States-1),
                ChannelCondition    INTEGER (0..127),
                MeanSchedulingDelay INTEGER (0 ..255),
                QoSIndicator        INTEGER (0..127),
                QoSIndicatorForecast {t, QoSIndicator}, {t+500ms, QoSIndicator},..
                LoadStatus          ENUMERATED {Low, Medium, High}
                LoadStatusForecast  {t, LoadStatus}, {t+500ms, LoadStatus},..
}
```

where *PhysCellId* and *TCI-StateId* are RRC parameters defined in [12], *ChannelCondition* may be indicated via SINR measurement of the serving cell and TCI state, sent through SINR-Range IE defined in [12]. The parameter *MeanSchedulingDelay* is computed by the UE, based on averaged time difference between scheduling request and UL transmission time indicated in the UL grant. This value may be mapped to an integer value. Subsequently, the MgtN gathers the QoE reports from all UEs and then aggregates and anonymises them into a SN-wide report. The aggregation and anonymisation method is MgtN implementation-specific. A simple method for achieving aggregation and anonymization of the UE reports at the MgtN side would be for the MgtN to average the collected information from the UEs. However, more sophisticated methods using deep learning could also be deployed at the MgtN side. Upon aggregation, the IE *QoESubnet* is created, which contains the aggregated QoE information.

```
QoESubnet ::=  SEQUENCE {
               PhysCellId                             INTEGER (0 ..1007),
               TCI-StateId                            INTEGER (0..maxNrofTCI-States-1),
               AggregatedChannelCondition             INTEGER (0..127),
               AggregatedMeanSchedulingDelay          INTEGER (0 ..255),
               AggregatedQoSIndicator                 INTEGER (0..127),
               AggregatedQoSIndicatorForecast         {t,  , QoSIndicator}, {t+500ms, QoSIndicator},..
               AggregatedLoadStatus                   ENUMERATED {Low, Medium, High}
               AggregatedLoadStatusForecast           {t, LoadStatus}, {t+500ms, LoadStatus},..
}
```

This anonymised IE, containing the aggregate QoE from the UEs of the SN for a specific cell ID and TCI state, will be provided to the recipient UE to assist it with its mobility decision. The final IE *QoESubnetAggregated* after aggregation and anonymization has the following form:

```
QoESubnetAggregated ::=    SEQUENCE {
               SubNetId        INTEGER (0 ..15),
               QoESubnet       QoESubnet,
}
```

After constructing the IE *QoESubnetAggregated*, the MgtN shares its own report with neighbouring MgtNs in periodic or event-driven fashion as shown in Step 4. Explicitly, the target of Step 4 is to provide the serving MgtN with QoE information from the neighbouring cells that the MgtN has not direct access, due to e.g., poor link quality at their cell edge.

3.2.4.2 Subnetwork-assisted Predictive Mobility

The overall process of SN assisted mobility is presented in Figure 28. Note that Steps 1-8 involving HO preparation and RRC reconfiguration are already present in 3GPP state-of-the-art as part of the CHO. The novelty lies in Steps 8-16 involving the “Capability Exchange” process and how the mobility information is assembled by the serving MgtN from neighbouring MgtNs and sent to the requesting UE.

As far as the “Capability Exchange” process is concerned, it could precede the HO Preparation process depending on the NW configuration. In Step 9, the UEs exchange their capabilities with their MgtN, indicating that they require SN support for mobility decision. During this step, the UEs may indicate to the MgtN the prepared cells along with which specific information elements are required for their mobility decisions. In Step 10, the serving MgtN1 sends a request to the neighbouring MgtNs indicating that it requires information from them. Note that the list of prepared cells, if provided by the UE, can be included in this request. Subsequently in Step 11, MgtN1 receives a confirmation from the neighbouring MgtNs whether they can support MgtN1 for UE1’s mobility decisions. Finally, MgtN1 informs UE1, whether it can provide support it for mobility decisions in Step 12.

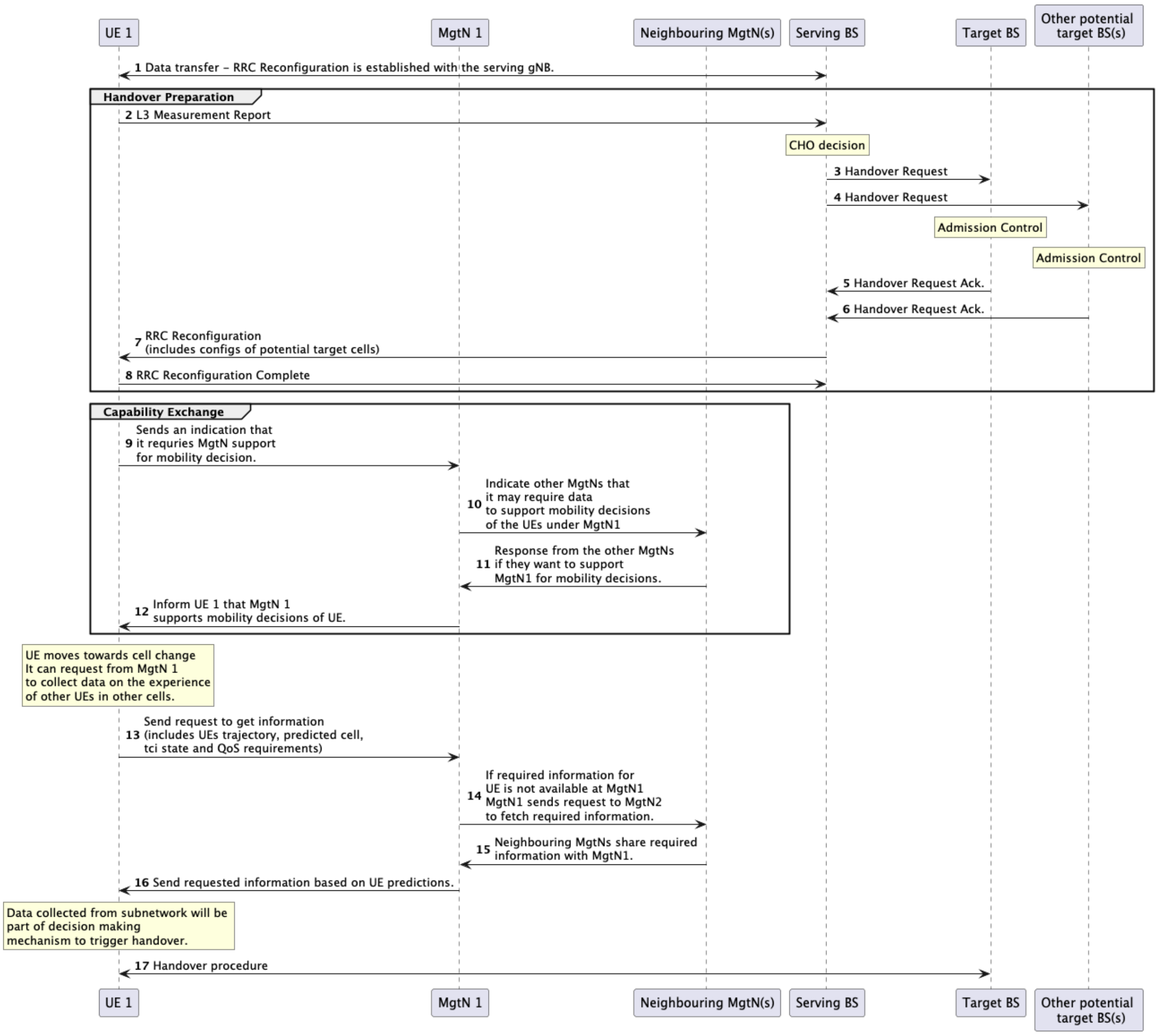

UE 1
MgtN 1
Neighbouring MgtN(s)
Serving BS
Target BS
Other potential target BS(s)
1 Data transfer – RRC Reconfiguration is established with the serving gNB.
Handover Preparation
2 L3 Measurement Report
CHO decision
3 Handover Request
4 Handover Request
Admission Control
Admission Control
5 Handover Request Ack.
6 Handover Request Ack.
7 RRC Reconfiguration (includes configs of potential target cells)
8 RRC Reconfiguration Complete
Capability Exchange
9 Sends an indication that it requries MgtN support for mobility decision.
10 Indicate other MgtNs that it may require data to support mobility decisions of the UEs under MgtN1
11 Response from the other MgtNs if they want to support MgtN1 for mobility decisions.
12 Inform UE 1 that MgtN 1 supports mobility decisions of UE.
UE moves towards cell change It can request from MgtN 1 to collect data on the experience of other UEs in other cells.
13 Send request to get information (includes UEs trajectory, predicted cell, tci state and QoS requirements)
14 If required information for UE is not available at MgtN1 MgtN1 sends request to MgtN2 to fetch required information.
15 Neighbouring MgtNs share required information with MgtN1.
16 Send requested information based on UE predictions.
Data collected from subnetwork will be part of decision making mechanism to trigger handover.
17 Handover procedure
UE 1
MgtN 1
Neighbouring MgtN(s)
Serving BS
Target BS
Other potential target BS(s)

Figure 28 Overall process of subnetwork-assisted predictive mobility for UE1 in Figure 26.

Step 13 is invoked in only if UE1 has received confirmation from the MgtN1 that it will support its mobility decisions in Step 12 and when UE1 detects poor link quality with its serving BS. During this Step, UE1 sends a request to MgtN1, i.e. its serving MgtN, to collect information about potential target BSs. UE1 request includes target/prepared cell IDs, TCI as well as measurements. Additionally, the UE may include its trajectory prediction along with the predicted target cell IDs. An example of the *Information Element* (IE) *QoEUERequest* sent by the UE to MgtN1 is as follows:

```
QoEUERequest ::=                            SEQUENCE {
            devId                           INTEGER (0 ..255),
            TargetCellTCIStateForecast      {t,   [PhysCellId,   TCI-StateId,   Measurement]},   {t+500ms,
                                            [PhysCellId, TCI,-StateId, Measurement]},..
            QoSIndicatorForecast            {t, QoSIndicator}, {t+500ms, QoSIndicator},..
            LoadStatusForecast              {t, LoadStatus}, {t+500ms, LoadStatus},..
}
```

Upon reception of UE1's request, MgtN1 checks whether it has information in its database about the QoE from other UEs of the forecasted [Cells, TCI StateId] indicated in the request. If this information is not available at the MgtN1 side, MgtN1 sends a request to its neighbouring MgtNs to fetch this information in Step 14. The neighbouring MgtNs reply by sending back to MgtN the requested

QoESubnetAggregated IEs, in case their database contains the requested IEs. Note that Steps 14 and 15 of Figure 28 are identical to the exchange of Step 4 in Figure 27.

Finally, after gathering the requested information MgtN1 sends it back to the requesting UE1 in Step 16, thus enhancing UE1's mobility decision. Explicitly, the side information provided to the UE can assist the UE in maintaining and possibly improving its QoS quality. By contrast, the baseline SotA 3GPP mobility can degrade the QoS quality, since it only considers reference signal quality, being oblivious to QoS quality, while selecting a new target cell. Additionally, as the UE is now informed regarding the QoE conditions in potential target cells, the probability of having its QoS constraints not met and triggering reselection is reduced. This in turn leads to a reduction of the UE's interruption time.

3.2.5 Summary

A complete framework for subnetwork mobility has been presented in this framework, where procedures for SN formation, registration and (re-)selection have been introduced. In terms of subnetwork formation, a decentralized scheme has been proposed for deciding which devices become MgtNs through negotiation without any network involvement. For this reason, a flexible set of MgtN capabilities has been introduced. Subsequently, this scheme has been extended to involve the 6G BS as a communication backbone for the sake of increasing the communication radius between the nodes. Finally, a SN registration process has been defined to make the parent NW aware of the subnetwork, so that the virtual connections of [4] are established.

In terms of SN (re-)selection, a UE-centric scheme has been proposed, where the UE selects an appropriate SN with the aid of the newly introduced SN capabilities. A mobility coordination mechanism among the SNs has also been introduced, where the serving MgtN propagates SN capabilities from neighbouring SNs, thus assisting in the UE's mobility processes.

Finally, a SN-centric mobility procedure has been introduced to assist the mobility of the UE in the parent NW. In this context, the serving MgtN propagates QoE reports from neighbouring cells, created either by the serving MgtN or by the neighbouring MgtNs. These QoE reports contain additional side information such as NW load that enhance the cell selection decision, thus reducing interruption time and improving QoS.

3.3 COORDINATION WITHIN SUBNETWORKS

3.3.1 Introduction

In the previous Section, examples of coordination within the SN and across SNs have been presented, such as MgtN-assisted SN re-selection and the SN support for predictive mobility. Additional enhancements are possible by exploiting the potential coordination mechanisms within and across SNs leveraging from the locality of the nodes. More specifically, a coordinated location update process is presented in Section 3.3.2, while a coordinated SN mechanism is introduced in Section 3.3.3 for enhancing the L3 measurements of each device.

3.3.2 Control Plane function offloading

In D4.2 [4] the concept of Control Plane (CP) offloading was introduced. In this subsection a CP offloading solution focusing on mobility scenarios is discussed, especially for the case of devices moving together as a group. In this case, the conditions and actions are very similar for the individual devices, since they are in proximity like in the platooning use cases. Figure 29 shows such a moving group, where each device has its own registration towards the 6G NW.

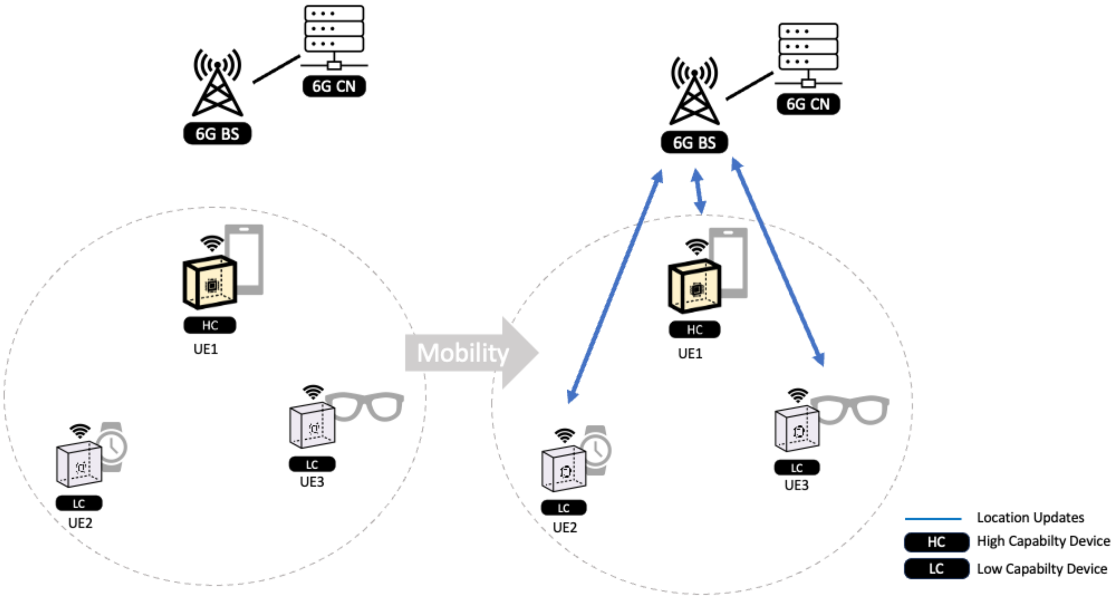


Figure 29 Individual Location Updates by nodes moving together

As shown in Figure 30, at each new location each device in Idle or Inactive state performs a Location Update (LU) which creates collisions on the communication resources as all the devices in the group compete for them. In particular, each device must wake up individually, perform a Random Access (RA) procedure, do connection establishment and then perform the corresponding messaging for LU procedures. This creates the following challenges:

1. Increased power consumption, which is relevant for LC devices that are battery constrained, such as wearables or devices with low battery levels.
2. Limited coverage and low efficiency of communication, which is relevant for LC devices that have limited communication capabilities, e.g. lower number of antennas/MIMO schemes, lower antenna performance due to physical/size limitations.
3. Increased NW load due to:
   - the uncoordinated nature of local devices performing Tracking Area Updates (TAU) or Radio Network Area Updates (RNAU) individually (e.g., when moving together into new Tracking Area (TA))
   - higher collision probability on RA channel, which increases procedural delays and affects MT services negatively (e.g. by being unreachable for a certain time)

   In general, this signalling load may even affect other devices in the cell.

Figure 30 Conflicts caused by Individual Location Updates

3.3.2.1 Proxy Location Updates

To overcome the challenges mentioned above, a rather simple solution would be that multiple UEs offload their CP functionality of performing LUs to a single device. Hence, all devices within the SN, like HCs, LCs or even Subnetwork Network Elements (SNEs) may offload that functionality to the HC node acting as a MgtN. As shown in the example in the left-hand side of Figure 31, UE2 and UE3 offload their UE Context related to the LU towards the UE1 which is acting as MgtN. The MgtN shall perform a Proxy LU on behalf of all the devices in the SN using their individual context information. The MgtN can then coordinate all upcoming location updates, thus avoiding conflicts and collisions, e.g., on RA channel, as illustrated in the right part of Figure 31.

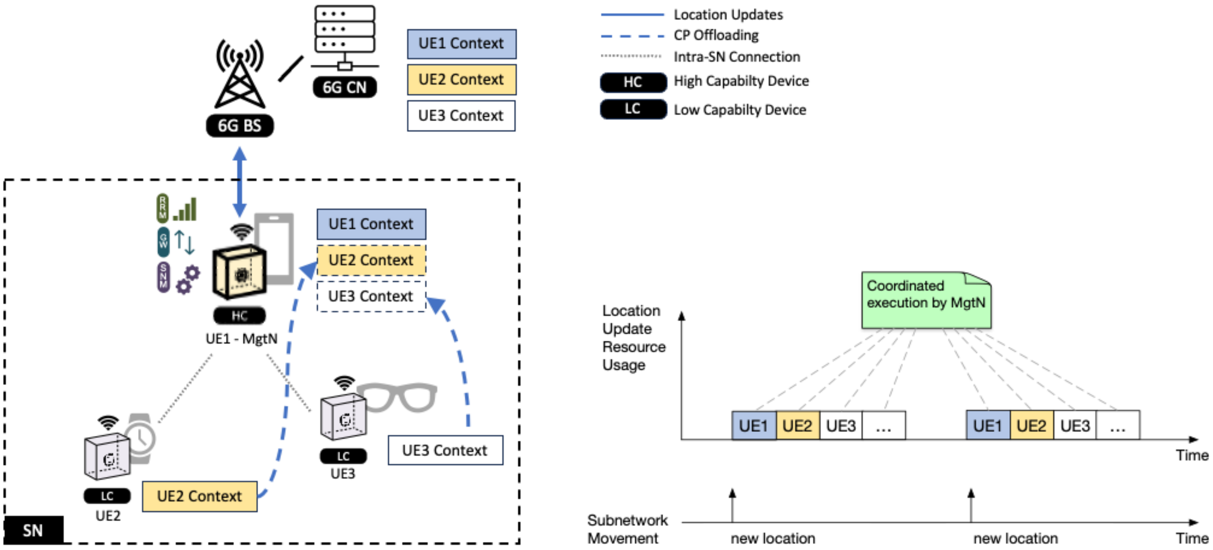


Figure 31 Location Updates by the MgtN

The challenges that come with this solution are mainly on privacy and scalability, since the MgtN needs to get access to the UEs' context (e.g. security context, UE ID, etc.) to perform the LU in a transparent manner towards the NW. The MgtN also needs to maintain a state machine per local device to perform the LU. It may also be obliged to perform hundreds of individual LUs, especially for large sub-networks, e.g. industrial networks, or consumer use cases with many personal devices. To address those challenges, more enhancements to the offloading scheme are required that also involve NW changes and are discussed further in the next subsections.

3.3.2.2 Batch Proxy Location Update

To improve the user privacy and the scalability of LUs at the MgtN, the NW and the offloading local devices shall exchange *Secured IDs*, e.g., during initial registration, for the purpose of sharing them with the MgtN. This is essential for avoiding sharing the whole context, as shown in Figure 32. When performing the LU, the MgtN shall use its own context and security credentials and shall only include the *Secured IDs* from the local devices, which offloaded LU by means of Batch Proxy LU. Only the NW can derive for which devices the Batch Proxy LU is directed without the necessity to reveal more information at the MgtN side. When responding to the LU, the NW may piggy-back additional UE specific information elements targeted to different *Secured IDs* to the message to the MgtN. Those can be encrypted using the UEs respective security context and therefore are transparent to the MgtN, as they can only be decrypted by the individual UEs itself.

Benefits

- Improved privacy, since:
  - Local devices' security context is not shared with the MgtN.
  - NW provides encrypted containers back to MgtN, which the MgtN shall forward to the local device, and which are transparent to the MgtN.
  - NW has no information about the formation and the nature of the SN; it only knows that all the indicated devices perform a LU together at a specific instance.
- Improved scalability, since:
  - Only a single connection needs to be established towards the BS; a single RA procedure, AS security procedure and LU procedure takes place, as indicated in Figure 32 right-hand side.

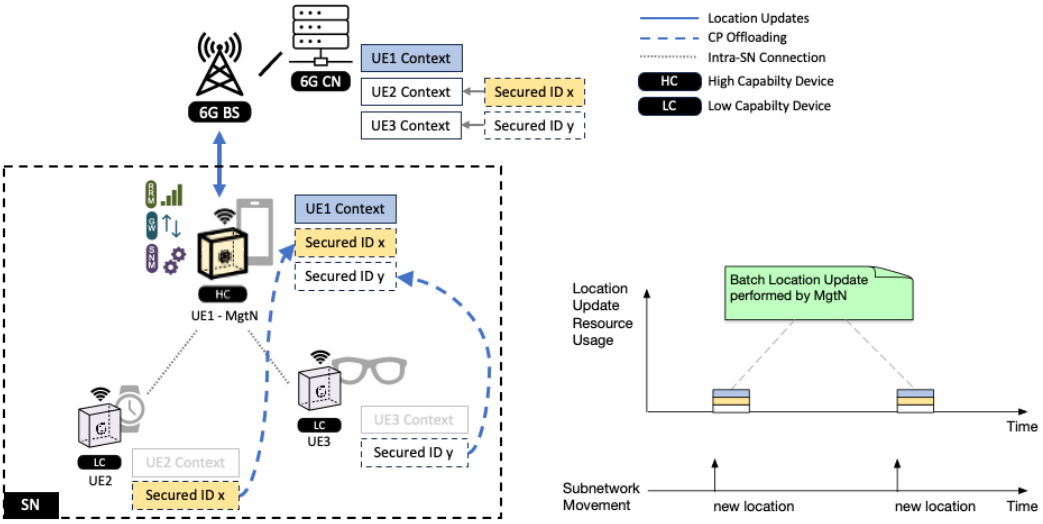


Figure 32 Batch Proxy Location Update

3.3.2.3 Group Proxy Location Update

Another option is to make the NW aware of the formation of a SN and keep it updated upon changes, such as when local devices are joining or leaving. Figure 33 shows how a Group ID could be assigned during the formation of this group or SN and how this Group ID is used in the LU to indicate the applicability of this Group Proxy Location Update for all group members. Similar to the approach described in Section 3.3.2.2, the NW can respond to individual UEs by adding protected IEs into the messages towards the MgtN using the UEs individual security contexts.

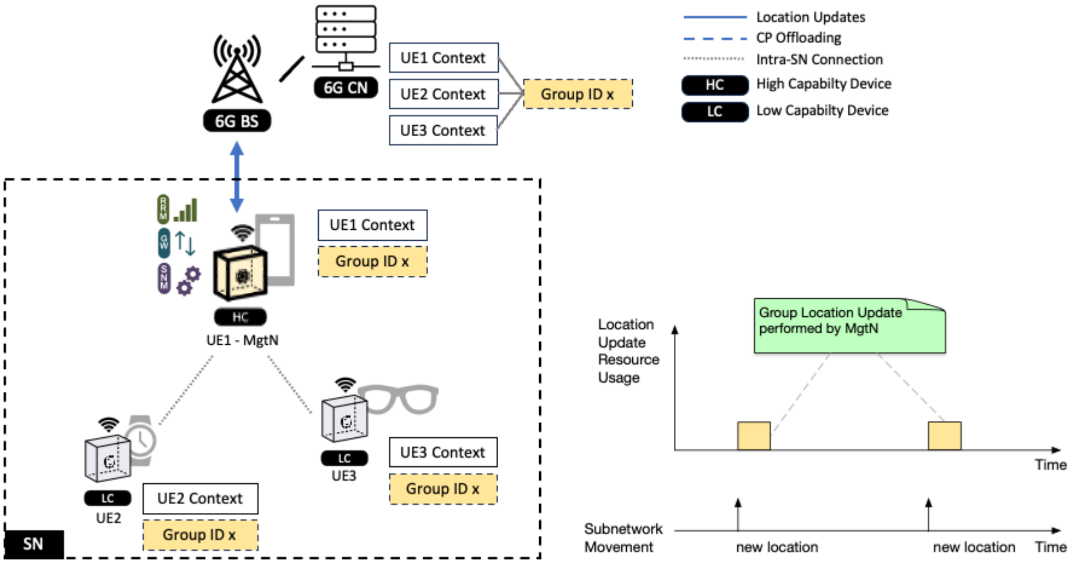


Figure 33 Group Proxy Location Update

3.3.3 Coordinated Measurements Framework for Subnetworks

Mobility in the current 3GPP framework [10] relies upon the UEs receiving measurement configurations from the BS via the Radio Resource Control (RRC) layer [12]. Based on the configurations where timings, quantities and reporting details are specified, the UEs shall perform the various measurements of reference signals from the serving cell and the neighbouring cells. In case of SL Relay, the UE is also configured by the network to perform measurement of reference signals from SL relays. Based on the above, it becomes evident that the current network-centric framework is more suitable to macro cells, since it assumes no spatial consistency on the channel conditions of the UEs. By contrast, SNs have their nodes located in close vicinity for all use-case categories [1][2]. Explicitly, the nodes in the same or neighbouring SNs experience similar channel conditions due to spatial coherence of the UE channels. This results in neighbouring UEs producing correlated reports, which is not leveraged by the current 3GPP framework. This in turn leads to unnecessary repeated functionality at both UE and the network sides.

Additionally, the new use case categories of D2.1 [1] and D2.2 [2], such as the smart factory and the immersive education, require the deployment of LC devices in terms of processing and power resources. Such devices would experience a significant increase in their power consumption [13] if they wake up on every *SSB-based RRM Measurement Timing Configuration* (SMTC) window to perform measurements. On the other hand, choosing not to wake up for power saving can result in a huge degradation of measurement performance. Hence, this severely impacts their mobility performance and leads to potentially more frequent Radio Link Failures (RLF) or out of coverage scenarios.

These two issues are visualized in Figure 34, where two neighbouring SNs are portrayed, each associated with a different BS. In terms of mobility, all devices HC, MgtNs or LC are configured by their BSs to measure the reference signals of both, the serving and the neighbouring BSs, along with the serving and neighbouring MgtNs. This also creates scalability issues in the core network side, especially in dense SN deployments.

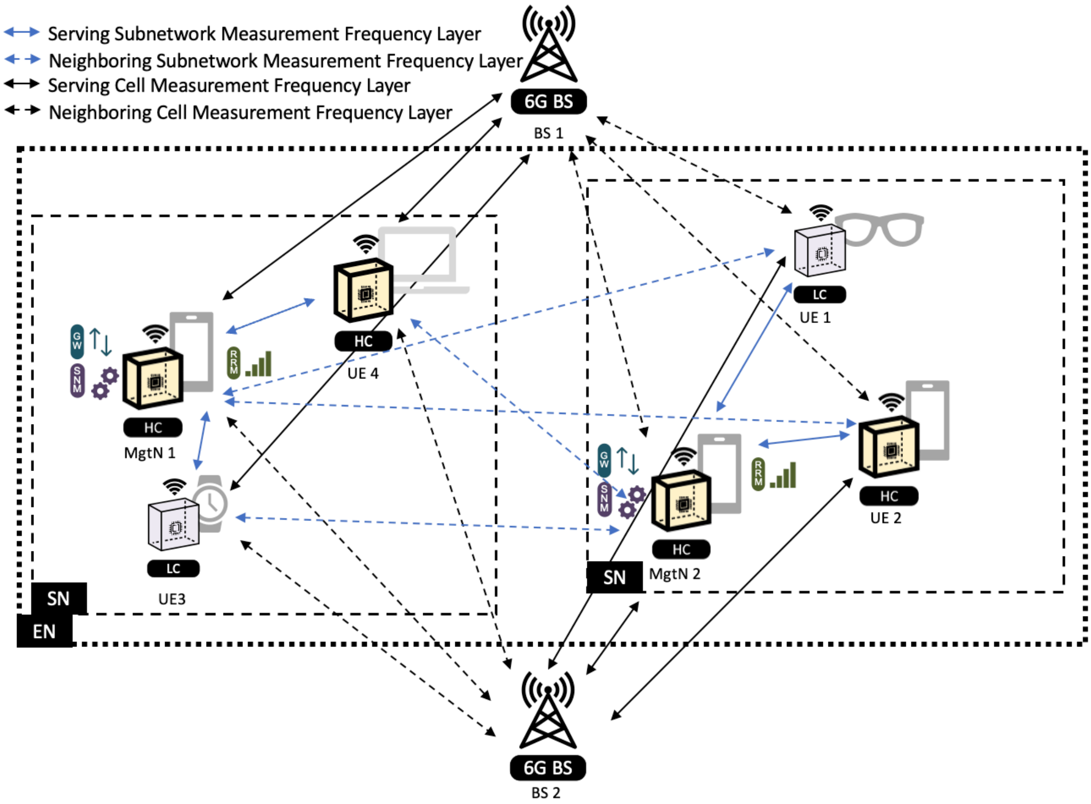

Serving Subnetwork Measurement Frequency Layer
Neighboring Subnetwork Measurement Frequency Layer
Serving Cell Measurement Frequency Layer
Neighboring Cell Measurement Frequency Layer
6G BS
BS 1
HC
UE 4
LC
UE 1
GW
SNM
RRM
HC
MgtN 1
HC
UE 2
LC
UE3
HC
MgtN 2
SN
EN
SN
6G BS
BS 2

Figure 34 Example of mobility measurements for the UEs of two neighbouring SNs.

One potential solution would be to exploit the spatio-temporal channel correlations to increase the measurement periods of the devices within the SN and particularly those of LC devices. In fact, there exist already studies in 3GPP [25], [26], which try to exploit the channel correlations. However, these models were designed for independent deployment at each UE and, thus, cannot exploit the spatio-temporal channel correlations and the possible synergies among neighbouring UEs.

With this background, a novel framework for coordinating the measurements within and across SNs is presented. This framework allows the neighbouring HC devices to share their measurement reports with other neighbouring devices through their MgtNs. The reports are then used to train models at each device enabling the devices to either infer or enhance their measurement reports with the aid of these models. For the LC devices, this results in power savings as they could avoid waking up their RF module and instead opt for inferring their measurement with the aid of their trained model. Additionally, all devices both LC and HC could enhance the accuracy of their measurements with the aid of their trained model. Note that the assumption has been made that the training process is more power efficient that the process of activating an RF power chain.

3.3.3.1 Coordinated Measurements Framework within the Subnetwork

The message sequence chart of the novel framework is shown in Figure 35. As a prerequisite, the nodes shall form a trusted SN. This assumption is crucial to levy privacy concerns, since the act of sharing their individual measurements with neighbouring nodes may be deemed a privacy concern. This trusted SN is formed e.g., by the devices of a single user or of users of the same group or family. Additionally, all nodes are configured by their serving MgtN and/or BS to perform measurements on both the serving and neighbouring MgtNs and/or BSs.

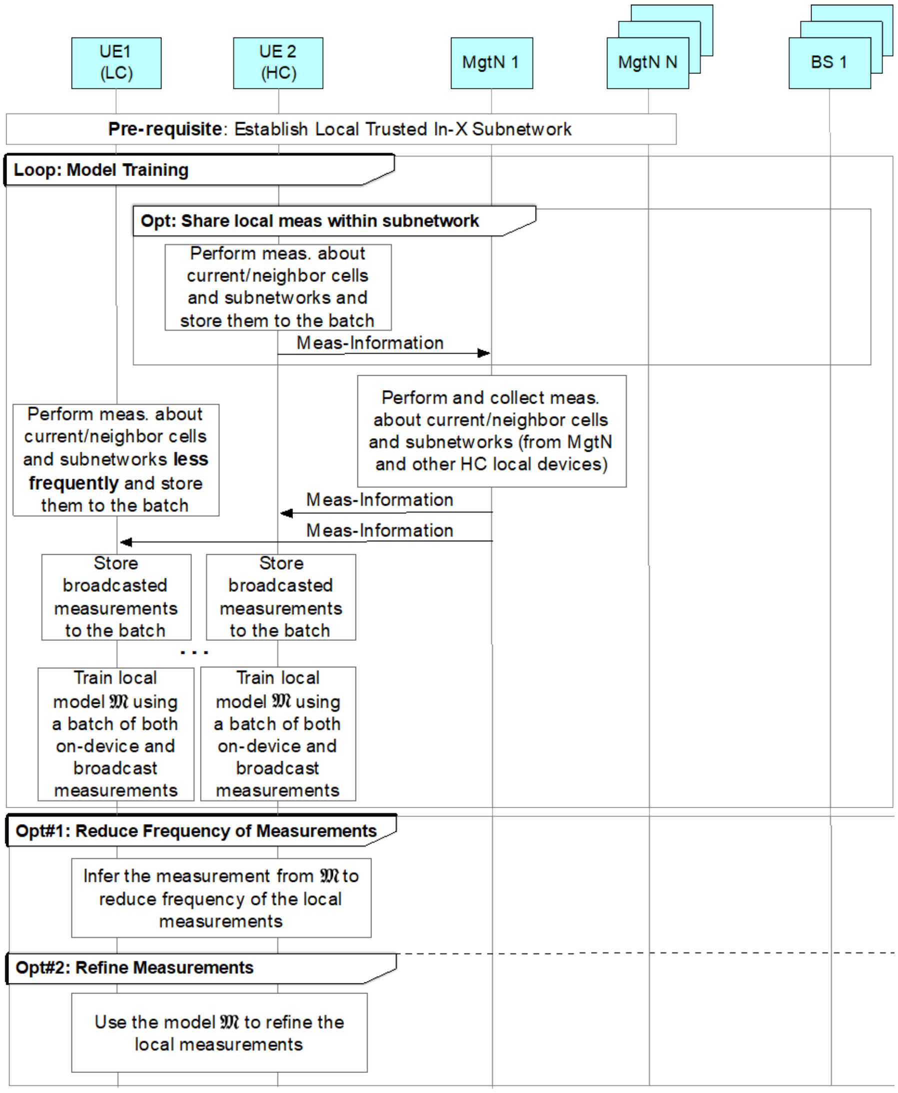

UE1 (LC)
UE 2 (HC)
MgtN 1
MgtN N
BS 1
Pre-requisite: Establish Local Trusted In-X Subnetwork
Loop: Model Training
Opt: Share local meas within subnetwork
Perform meas. about current/neighbor cells and subnetworks and store them to the batch
Meas-Information
Perform and collect meas. about current/neighbor cells and subnetworks (from MgtN and other HC local devices)
Perform meas. about current/neighbor cells and subnetworks less frequently and store them to the batch
Meas-Information
Meas-Information
Store broadcasted measurements to the batch
Store broadcasted measurements to the batch
...
Train local model 𝔐 using a batch of both on-device and broadcast measurements
Train local model 𝔐 using a batch of both on-device and broadcast measurements
Opt#1: Reduce Frequency of Measurements
Infer the measurement from 𝔐 to reduce frequency of the local measurements
Opt#2: Refine Measurements
Use the model 𝔐 to refine the local measurements

Figure 35 Message sequence chart of the coordinated measurements framework within the subnetwork.

Moving on to the model training process of Figure 35, both the HC and the MgtN devices perform regular measurements on both the serving and neighbouring MgtNs and/or BSs. The LC devices also perform the same measurements albeit at an increased measurement periodicity. Over time, these measurements are collected into a batch. The batches of the HC are then transmitted to the MgtNs, which in turn aggregate these batches from all the served HC devices as well as its own measurements. The MgtNs then transmit their aggregated batch of measurements to all devices served by their SN. This process is repeated periodically using e.g., a dedicated SIB via data channels. Subsequently, the devices utilitize these batches to train their individual models $\mathfrak{M}$. They then use their models for:

- **Reducing measurement frequency (Opt.1):** the UEs and especially the LC nodes opt for measuring less frequently by inferring their measurement from their model $\mathfrak{M}$. This results in activating their RF chains less frequently providing significant power savings.
- **Refining their measurements (Opt.2):** higher capability nodes can use their model $\mathfrak{M}$ along with their instantaneous measurements to refine the latter. This can improve the accuracy of the instantaneous measurements, since refinement acts like filtering thus suppressing noise.

3.3.3.2 Model Training Flow

Before delving into the specifics of the model training and its flow, the associated notation needs to be defined:

- MgtN includes in the broadcasted measurement information a list of measurements collected from different N devices (i.e., either MgtNs or local HCs) in the local network.
- M_n is the measurements conducted by the n-th device out of N devices in total and can be defined as follows: $$M_n = \left[X^{(n)}_{BS1-beam1}, X^{(n)}_{BS1-beam2}, \dots, X^{(n)}_{MgtN1-beam1}, X^{(n)}_{MgtN1-beam2}, \cdots\right],$$ where $X^{(n)}$ can be RSRP, RSRQ, SINR or a combination of the aforementioned metrics measured at the n-th device. The devices may time filter their measurements prior to sending them to the MgtN.
- $Y^{(i)}$ is the measurement conducted by the i-th UE and can be defined as follows: $$Y^{(i)} = \left[X^{(i)}_{BS1-beam1}, X^{(i)}_{BS1-beam2}, \dots, X^{(i)}_{MgtN1-beam1}, X^{(i)}_{MgtN1-beam2}, \cdots\right],$$ The UE may also filter its own measurement prior to feeding them in the model.
- $\mathfrak{M}^{(i)}$ is model of the i-th UE and it is comprised by a set of weights as follows, i.e. we have $\mathfrak{M}^{(i)}\left(W_1^{(i)}, \dots, W_J^{(i)}\right)$.
- $\hat{Y}^{(i)}$ is denoted as the inferred measurement by the $\mathfrak{M}^{(i)}$ model.

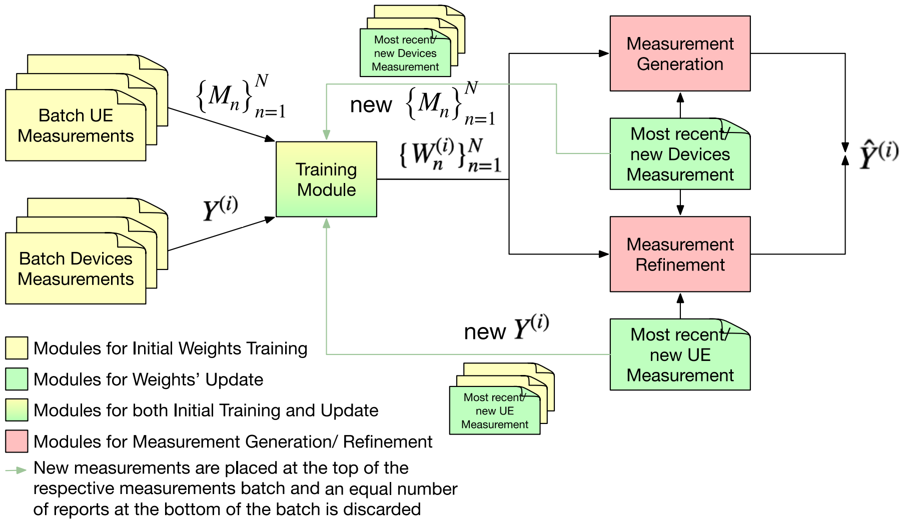

Most recent/ new Devices Measurement
Batch UE Measurements
$\{M_n\}_{n=1}^{N}$
new $\{M_n\}_{n=1}^{N}$
Training Module
$\{W_n^{(i)}\}_{n=1}^{N}$
$Y^{(i)}$
Batch Devices Measurements
Measurement Generation
Most recent/ new Devices Measurement
Measurement Refinement
$\hat{Y}^{(i)}$
new $Y^{(i)}$
Most recent/ new UE Measurement
Most recent/ new UE Measurement
Modules for Initial Weights Training
Modules for Weights' Update
Modules for both Initial Training and Update
Modules for Measurement Generation/ Refinement
New measurements are placed at the top of the respective measurements batch and an equal number of reports at the bottom of the batch is discarded

Figure 36 Model training and update flow for a single LC or HC node.

Based on the above, the model $\mathfrak{M}$ can be as simple as a linear model or more complex, such as a deep neural network. Naturally, the target of the module training is to evaluate the weights $W_1^{(i)}, \dots, W_J^{(i)}$ such that the mean square error between inferred measurement values $\hat{Y}^{(i)}$ and the input measurement is minimized. The flow for each device is shown in Figure 36. At first, each of the UEs accumulates batches of both neighbouring devices measurements as well as batches of its own measurements. Note that these batches are collected over an extended time, which often spans over multiple measurement periods. This is crucial for the initial model training which requires often a significant number of observations. Subsequently, the model is trained by evaluating the set of the initial weights is $W_1^{(i)}, \dots, W_J^{(i)}$. This process is repeated for fine-tuning and updating the model, when new batches of neighbouring node measurements are received by the UE and it performs new measurements, as shown by the green arrows of Figure 36. Using the trained model $\mathfrak{M}^{(i)}\left(W_1^{(i)}, \dots, W_J^{(i)}\right)$ the UE can produce its output measurement $\hat{Y}^{(i)}$ by inferring using the model. This output measurement can also be combined with a recent UE measurement to enhance the latter's accuracy. This process is visually portrayed in Figure 37, where the UE has as input a batch of neighbouring UE measurements. These batches are received with X periodicity. Using a batch, the UE produces its inferred measurement $\hat{Y}^{(i)}$. Additionally, the UE may continue to measure and produce its own reports $Y^{(i)}$ albeit with increased periodicity. To enhance the measurement outcome, the UE could combine the inferred and the instantaneous measurement in Opt.2 with a weighted average.

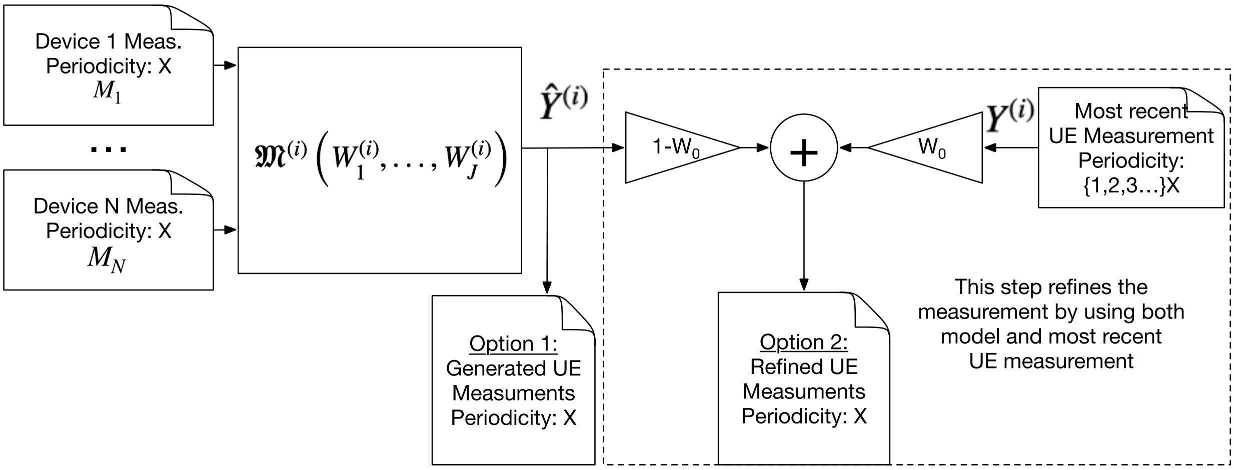

Device 1 Meas.
Periodicity: X
M_1
...
Device N Meas.
Periodicity: X
M_N
$\mathfrak{M}^{(i)}\left(W_1^{(i)},\ldots,W_J^{(i)}\right)$
$\hat{Y}^{(i)}$
1-W$_0$
+
W$_0$
$Y^{(i)}$
Most recent UE Measurement Periodicity: {1,2,3…}X
Option 1: Generated UE Measuments Periodicity: X
Option 2: Refined UE Measuments Periodicity: X
This step refines the measurement by using both model and most recent UE measurement

Figure 37 Flowchart for demonstrating how the UE can enhance their measurements with their model in Opt.2.

3.3.3.3 Signaling for Propagation of Measurements across the Subnetwork

As far as the measurement reports from devices to the MgtN are concerned, RRC *Information Element* (IE) *MeasResults [12]* could be readily used to map each measurement M_n into a message. However, at the UE side the model needs to distinguish the source of each report in order to benefit from the spatio-temporal correlation. For this specific reason anonymized device IDs to distinguish the origin of the measurement report are included after aggregation at the MgtN side. Note that these IDs are only relevant for measurement sharing and have no connection to other IDs which would yield the real identity of the source HC UEs. This is essential for training the model by ensuring that the individual input reports are well distinguished and correlated in time. Another important information would be the inclusion of the mobility state of the device This is helpful both for the MgtN e.g., to propagate the measurements of stationary UEs less frequently, and for the UE models as well. Based on the above the IE *MeasResultsDev* is introduced, it can be defined as follows:

```
MeasResultsDev ::=  SEQUENCE {
            devID             INTEGER (0 ..255),
            subNetId          INTEGER (0 ..15),
            MeasResults       MeasResults,
            MobilityState     ENUMERATED {stationary, normal, medium, high}          OPTIONAL
}
```

where the parameter *devID* corresponds to an anonymized device ID, unique for each SN device, the parameter *subNetId* is the anonymized SN ID, which is unique for each SN, the parameter *MeasResults* corresponds to the respective RRC parameter for the measurement results of each device, as defined in [38.331]. Finally, the parameter *MobilityState* is optional and corresponds to the mobility state of the device. Subsequently, MgtNs aggregate the IE from *MeasResultsDev* into a single message, so that the latter is broad- or multicast across the SN as follows:

```
MeasResultsSubnet ::=  SEQUENCE (1 .. numDevices) OF MeasResultsDev
```

As already mentioned, the IE *MeasResultsSubnet* is periodically transmitted to the UEs across the SN using a dedicated SIB via the data channel.

3.3.3.4 Inter-subnetwork Collaboration

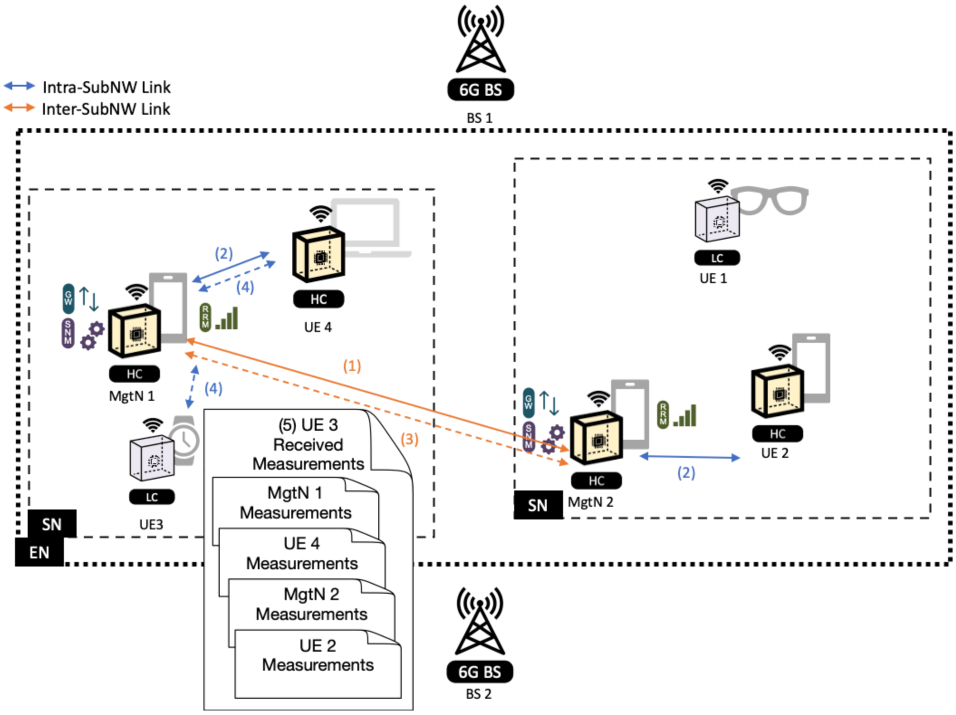


Figure 38 Inter-SN sharing of HC UE measurements

The concept mentioned in Section 3.1.3 - where the MgtNs can discover neighbouring MgtNs, perform mutual authentication and establish trusted and secure connections - can be readily applied to share the reports across neighbouring SNs. This process is portrayed in Figure 38, where there are two neighbouring SNs. More specifically, the LC UE3 can acquire the measurement reports from the HCs UEs of both SNs as follows:

1. MgtN1 and MgtN2 establish a secure and trusted connection, which is referred to as “Inter-SN link”
2. The HC UE4 transmits its report to MgtN1, and the HC UE2 to MgtN2.
3. MgtN aggregates the reports received (in case of more HC UEs) and transmits the aggregated reports through the inter-SN link to MgtN1.
4. MgtN1 aggregates the reports received from within the SN as well as those from MgtN2 and transmits them to LC UE3.
5. UE3 has now the report from all HC devices within the SNs managed by MgtN1 and MgtN2.

Since the available measurements are increased in this way, the model becomes more accurate and is not confined within a single SN. Note that the new measurements are also spatio-temporally correlated due to the proximity of the neighbouring SNs.

3.3.3.5 Benefits

The coordinated measurements framework enables LC UEs to have more accurate measurements, while achieving significant power savings from the increased measurement period. Note there is a reasonable assumption that the training of the models is less energy demanding than performing frequent measurements in the SMTC window. In terms of signalling, there is a slight increase due to the SIB containing the aggregated report; however, aggregating multiple measurements (over time and frequency) and signaling them via SIB is expected to outperform individual measurements. In Figure 39, the LC UEs after the initial training of the model can enter the reduced measurement frequency mode where they could measure every e.g., 2 SMTC windows. In this case the paging duration might be increased slightly due to the reception of the new SIB message with the reports. As for the HC UEs, the associated overheads are a slight increase in the SMTC window duration to send the report to the MgtNs as well as in Paging duration to receive the new SIBs.

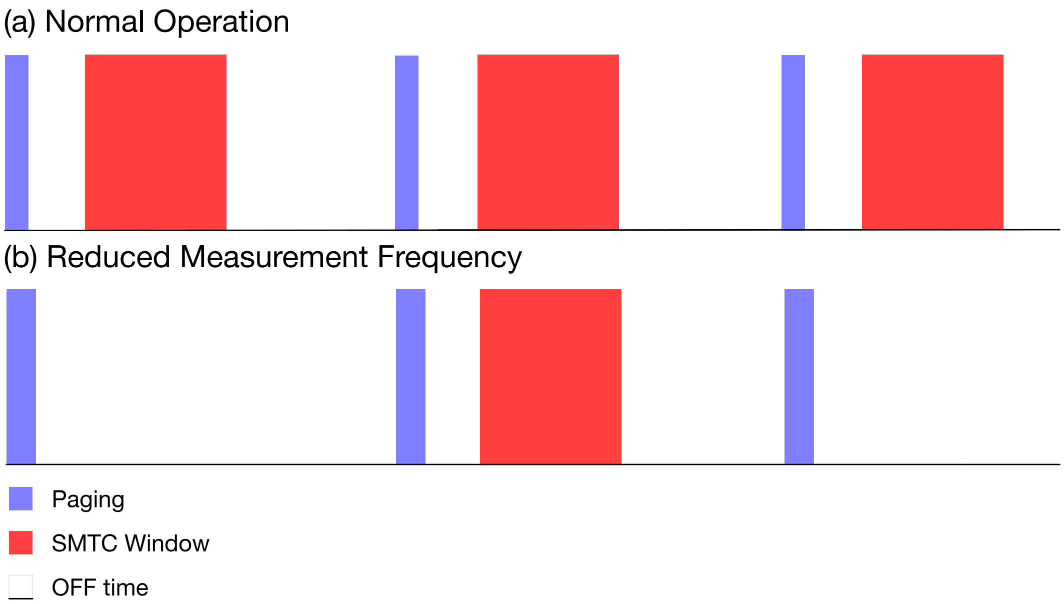


Figure 39 Visual portrayal of the reduced measurement frequency ON time vs the normal 3GPP operation.

3.3.4 Summary

In this Section, coordination mechanisms within the SN and across SNs have been proposed, which improve the performance of the individual devices. More specifically, three novel LU mechanisms have been proposed in Section 3.3.2, namely the proxy location update, the batch proxy update and the group proxy update. All three solutions achieve reduction in collisions from individual LU attempts, compared to the case where there is no coordination. The batch update and the group proxy update manage to also reduce the signalling overhead, as in both cases the MgtN performs the LU on behalf of a group of UEs. The group proxy update has increased privacy and security, since there is no need for sharing UE contexts due to the introduction of the “Group ID”, which is linked by the 6G BS to the UE contexts of the UEs participating in a specific SN.

As far as the coordinated measurement framework for SNs is concerned, a SN process has been introduced for sharing the L3 measurements of HC nodes within and across the SNs. This process enabled UEs within the SN to train their local models for the sake of inferring their L3 measurements. at the cost of a small signalling overhead. The inferring of the measurements yields power savings since the LC UE can extend their DRX cycle as they now do not have to measure on every SMTC window.

3.4 QUALITY OF SERVICE ASPECTS

3.4.1 Introduction

In 3GPP, the Quality of Service (QoS) framework [15] ensures end-to-end performance guarantees for specific applications and services, both for downlink and for uplink. The QoS framework is based on QoS flows, which present the finest granularity of QoS treatments in the 5G network.

With the growing interest in immersive experiences, including VR headset and haptic sensors, the demand in timely delivery of each packet increases, putting higher requirements on the currently defined QoS framework in general, but also to handle QoS aspects related to subnetworks.
This demand includes support for so called multi-modality where several traffic flows related to the same application has a dependency, i.e., traffic related to e.g., video, audio and haptic feedback.

The QoS framework is currently being enhanced in 3GPP with basic support for multi-modality information related to QoS flows, but this is not sufficient to handle delay and time critical dependencies between packets related to different flows, and the concept of multi-modality has not been introduced for any type of relay operations. Enhancements of the existing solutions are needed to support Multi-modal QoS flows, within and between subnetworks or parent networks. This is very relevant for immersive XR traffic use cases, particularly for the use case where a relay node is needed, such as in the subnetwork scenarios described in this project and exemplified in this delivery.

Furthermore, this demand for support of multi-modal traffic also puts higher requirements on scheduling of resources for data transmission. All devices related to the subnetwork should get the same treatment of resources irrespectively of being uplink or sidelink resources. For this the scheduler, e.g. the base-station benefits from having full and overall information from all devices transmissions needs within the subnetwork.

3.4.2 QoS dependencies for Multi-modal streams

In the following section, proposals to the QoS framework for In-X subnetwork will be described. The framework can be applicable to any use-cases described in [2]. However, in this section, we focus on the consumer subnetwork use-cases, particularly the use-cases related to XR immersive experience, such as indoor interactive gaming The general and high-level requirements related to XR applications have been described in the previous deliverable [4]. Supporting immersive experience is one of the applications that have been mentioned in [67].

With continuously ongoing progress and development in man-machine interfaces, interactive and high resolution XR applications, including VR headset and haptic sensors and actuators creates a demand to offer an even stronger immersive experience. Furthermore, the future of multimedia and human centric communication will provide a real time interactive and immersive experience, including holographic telepresence, useful in different scenarios, like interaction at remote working, social interactions, entertainments, education, remote live performance and remote interactive gaming. Numerous technology trends and enablers are also listed, including:

- AI-native air interface,
- Integration of sensing and communication,
- Device-to-device wireless communication with extremely high throughput,

- Ultra-accuracy positioning and low latency.
- Other technologies (spectrum utilization, low power consumption and shared accurate time and frequency information within networks).

Enhancements of the radio network are expected and needed to support challenging applications, such as the immersive experience application as mentioned above. The enhancements here are foreseen to handle various QoS requirements from multiple users to provide a flexible and reliable user experience. This may lead to a reformed RAN architecture and optimized radio protocols, interworking between terrestrial and non-terrestrial and ultra-dense networks, where a dense deployment of transmission points (TRPs) can support user experience with high QoS requirements in a spectrum, energy, capacity and coverage efficient manner.

3.4.2.1 Representative Use-case: Indoor Interactive Gaming

In our investigation, we select indoor interactive gaming (consumer subnetwork) as the use-case to develop the QoS framework for the subnetwork operation. This use-case has stringent requirements in terms of data rates, latency, and synchronization which could be benefited by the operation of subnetwork. In Figure 40, the traffic flows within the subnetworks are illustrated. The red and green lines are representing the link for downlink and uplink direction, respectively that are being relayed by the MgtN/HC. Furthermore, these links are between HC/MgtN as the access-point/relay node with network functions (e.g., gateway, computation offloading, RRM) to the HC and/or LC devices in the respective subnetwork. The link between the access-point/relay node, HC device, to the 6G parent network is represented in dash-purple line. Lastly, the link to/from SNE is represented in thin-blue line. All of the links may have different link characteristics, depending on the type of traffic that will occur e.g., traffic from sensor to MgtN, traffic between different MgtNs, or aggregated traffic between subnetwork and parent 6G network.

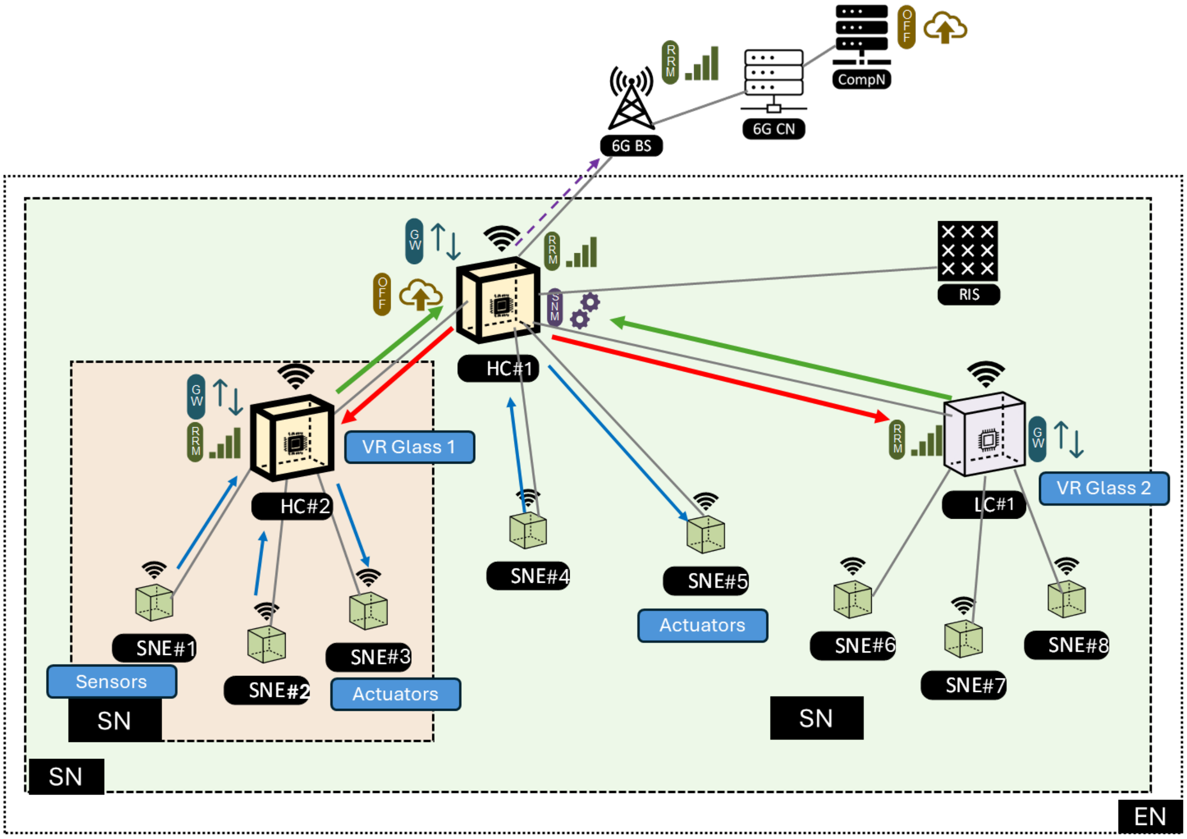


Figure 40 Subnetworks architecture and the traffic flows of indoor interactive gaming

The various traffic types can be video, data, and pose control. The HC#2 can represent the VR glass where the device can perform video rendering by itself. The HC#2 can also act as a gateway to the sensors, controllers, and actuators attached to the users. There can be other users in the room with similar HC or LC device. In case of LC#1, we consider the device performs split rendering at the computation node. The computation node here can be the HC#1 that has computation function. The computation node could receive the computation task, perform the task, and produce the task to the intended node. All the devices and subnetwork elements are connected to an HC#1 which carries gateway function, RRM, and computational offload. We consider all those network elements are relatively close, or such as within the target 6G-SHINE use cases (i.e., 10 m). The primary challenge of this use case is to support high data rate in a timely manner and with synchronized data delivery.

3.4.2.2 Existing Solutions elated to QoS and Relay

To support the mentioned use case above, and also many other use cases and scenarios discussed in this 6G-SHINE project as described in [2], QoS service aspects and requirements become very important to consider when designing even more advanced and complex systems involving several and different elements, including a subnetwork architecture.[4].

In 3GPP, concepts related to Quality of Service have been specified. The current QoS concept up to Rel-17, is based on that the UPF in the Core Network sends and receives the data with different QoS flows. Each flow has its specific QoS requirements such as the priority, allowed delay and the packet error rate (PER).

A QoS flow contains data where the QoS is described by a parameter 5QI (5G QoS Identifier) which identifies the QoS characteristics of that QoS flow [18]. Based on the 5QIs, RAN node can prioritize the data in the different QoS flows when scheduling transmissions over the air interface in order to fulfil the required QoS of each QoS flow.

In 3GPP Rel-18, the QoS concept for XR traffic is adapted with PDU Set handling where the PDU Set QoS parameters, PDU Set Delay Budget (PSDB), PDU Set Error Rate (PSER) and the PDU Set Integrated Handling Information (PSIHI) are added to the description for a QoS flow.

Network Relays were introduced in 3GPP Rel-17, including UE-to-Network Relays. An enhancement is introduced in Rel-18 by supporting UE-to-UE Relay. The QoS concept is now also valid for the Relays where the sidelink QoS characteristics are identified by the PQI which is similar to the 5QI and defined for the device-to-device interface. Thereby the QoS concept in 3GPP covers all UE to NW communications with or without relays. It also covers the corresponding UE to UE communication. So, the QoS framework is already defined, including sidelink relay. This can be used as basis to support the concept of subnetworks, even if not all required functionalities are at hand (e.g. PDU set handling).

Furthermore, in 3GPP Rel-19, the concept of multi-modality was discussed and eventually some basic parts were included in the revised work item description for XR [66], where Core Network can indicate to the base stations which QoS flows have a Multi-modal relation. It is up to base station implementation to handle this information, for example how to take into account when to perform the scheduling of traffic related to the different QoS flows.

There are two types of relays defined in 3GPP: Layer 2 relay and Layer 3 relay. In a Layer 2 relay the data is handled just above the RLC in the protocol stack as shown in Figure 41. Here the Sidelink Relay Adaptation Protocol (SRAP) is added in the relay and the SDAP and PDCP layers are terminated in the remote UE. Thereby the QoS flows in Layer 2 relays are defined between the gNB through the relay to the remote UE. [10].

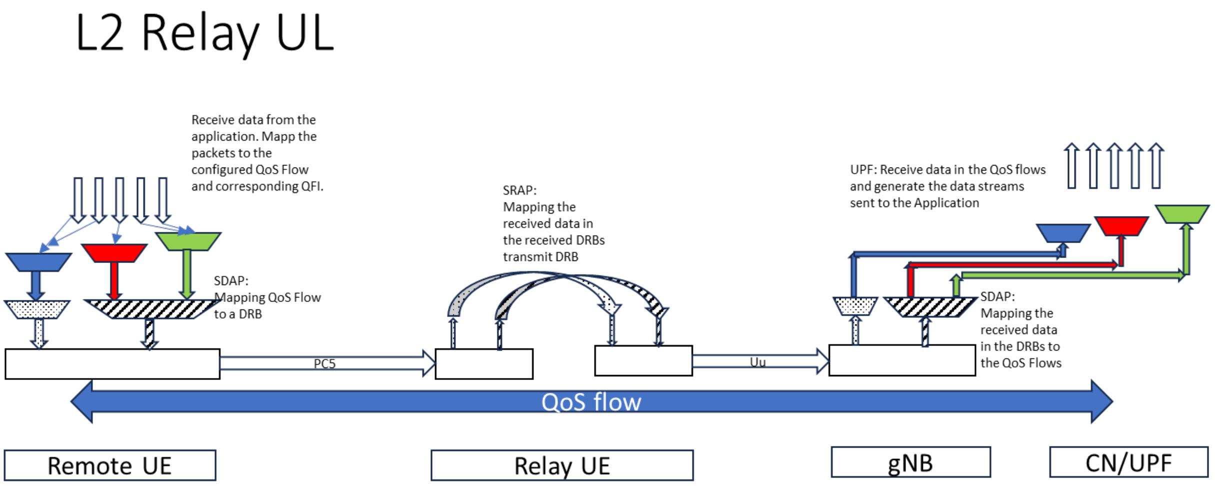

L2 Relay UL
Receive data from the application. Mapp the packets to the configured QoS Flow and corresponding QFI.
SDAP: Mapping QoS Flow to a DRB
SRAP: Mapping the received data in the received DRBs transmit DRB
UPF: Receive data in the QoS flows and generate the data streams sent to the Application
SDAP: Mapping the received data in the DRBs to the QoS Flows
PC5
Uu
QoS flow
Remote UE
Relay UE
gNB
CN/UPF

Figure 41 The handling of the QoS flows in an U2N L2 Relay

In a Layer 3 relay the relaying functionality is handled in a new Proximity-based Services (ProSe) Layer above the RAN layers and the QoS flow is terminated in the Relay as shown in Figure 42. A new set of QoS flows are configured for the communication between the Relay and the remote UE [9].

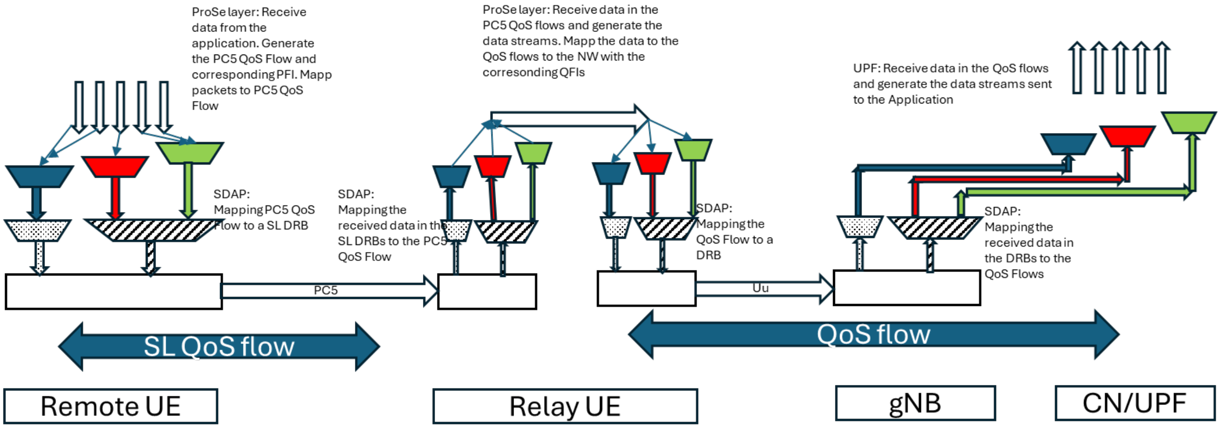

ProSe layer: Receive data from the application. Generate the PC5 QoS Flow and corresponding PFI. Mapp packets to PC5 QoS Flow
ProSe layer: Receive data in the PC5 QoS flows and generate the data streams. Mapp the data to the QoS flows to the NW with the corresonding QFIs
UPF: Receive data in the QoS flows and generate the data streams sent to the Application
SDAP: Mapping PC5 QoS Flow to a SL DRB
SDAP: Mapping the received data in the SL DRBs to the PC5 QoS Flow
SDAP: Mapping the QoS Flow to a DRB
SDAP: Mapping the received data in the DRBs to the QoS Flows
PC5
Uu
SL QoS flow
QoS flow
Remote UE
Relay UE
gNB
CN/UPF

Figure 42 The handling of the QoS flows in an U2N L3 Relay

For a subnetwork using UE to NW (U2N) sidelink relay, where the communication between the base station and the device, here called a remote UE is relayed over a UE-to-Network relay, the QoS handling is different depending on the type (L2 or L3) of relay which is used.

For a Layer 3 relay, the data is handled above the SDAP layer (i.e., in the ProSe layer), particularly in the relay UE where the QoS flow is terminated. A new QoS flow defined for the sidelink (PC5) communication with a new identifier PQI, (PC5 QoS Identifier) [9] is used in a similar way as 5QI is used for the Uu interface. So the end-to-end QoS is handled by two separate QoS flows with different identifiers, 5QI and PQI.

For Layer 2 relays, the QoS Flow defined by the 5QI is used over both the Uu interface and the PC5 interface. In practice, it is up to the base station implementation to handle the QoS handling over the

Uu and PC5 interfaces. For the case of UE-to-UE Relay, it is up to the relay node implementation to handle the QoS split [10].
For a Subnetwork element or entity there is no solution defined on how to handle and enforce QoS requirements for scenarios where multiple entities in subnetworks are involved. This holds also true when it comes to the support of multi-modal traffic streams.

3.4.2.3 Issues & Challenges on supporting QoS for subnetworks

The QoS framework defined in 3GPP with the QoS flow concept and with basic support for multi-modality indication an "MMSID" related to a certain QoS flow is not sufficient to handle delay and time critical dependencies between packets related to different flows.

Furthermore, the existing QoS framework that has been defined in 3GPP is for UE-to-Network and UE-to-UE Relays as described in the previous section. The concept of multi-modality has not been introduced for any type of relay operations.

Enhancements of the existing solutions are needed to support multi-modal QoS flows, within and between subnetworks or parent networks. This is very relevant for immersive XR traffic use cases, particularly for the use case where a relay node is needed, such as in the subnetwork scenarios described in this project and exemplified in this delivery.

Currently the QoS handling and split of e.g., packet delay budget (PDB) to support the QoS requirements over relay is done by the base station, or by the relay. In order to introduce support for autonomously controlled subnetworks and/or hierarchical structures that may even be consisting of multiple relay nodes, some or all of the QoS logic may have to be handled on a subnetwork level. The big challenge is how to handle and route traffic with packets belonging to different modal streams originated from different sources and potentially intended for different recipients. For example, the traffic originated in different game engines creating video and audio output and haptic sensor output and deliver to different VR headset and motion sensors e.g., residing in different devices or so called subnetwork elements (i.e, HC, LC, and SNE). In addition to the different QoS streams need to be treated and routed to the correct SN entity while the relation between the QoS streams needs to be considered. Furthermore, concurrent traffic flows can serve a different number of recipients, some traffic is intended for many elements, and other traffic is intended for a single or only a few elements. The elements can reside within the subnetwork, or be part of another subnetwork, but traffic can be within or between subnetworks, or between elements outside the parent network, where the subnetwork needs to communicate traffic via the 6G network

Different challenges and requirements may reside depending on use case and scenarios, but the solutions should be valid not only for the mentioned gaming use case but also for other use cases discussed and investigated in the 6G-SHINE project. These are industrial use cases, e.g., robot control, with extreme requirements on latency, for example operating quick movements of a robot arm, or automotive use cases, supporting a potential wireless vehicle subnetwork which need to ensure that control functionality depending on sensors and actuators is able to operate under all conditions at low latency, fulfilling all safety requirements from the car industry.

There is for above reasons and background a need to introduce new mechanisms in and for subnetworks, where e.g., a management node responsible for the subnetwork, and for interacting with the parent network or other subnetworks, can hold functionality to manage the QoS policies and the traffic within and between subnetworks and parent network in an efficient manner, enforcing the latency and time alignment requirements to uphold the synchronization of data packets belonging to different QoS flow or modal-streams to keep a synchronized data delivery for an immersive experience.

3.4.2.4 QoS framework enhancements for various data flow scenarios.

In this section we describe four different scenarios where solutions for the QoS handling of multi-modality for In-X subnetworks are presented and including the proposed enhancements. The scenarios are based on Figure 43 below, where the communication is between the application function (AF) in the network and any of the elements in these subnetworks or between any of the elements.

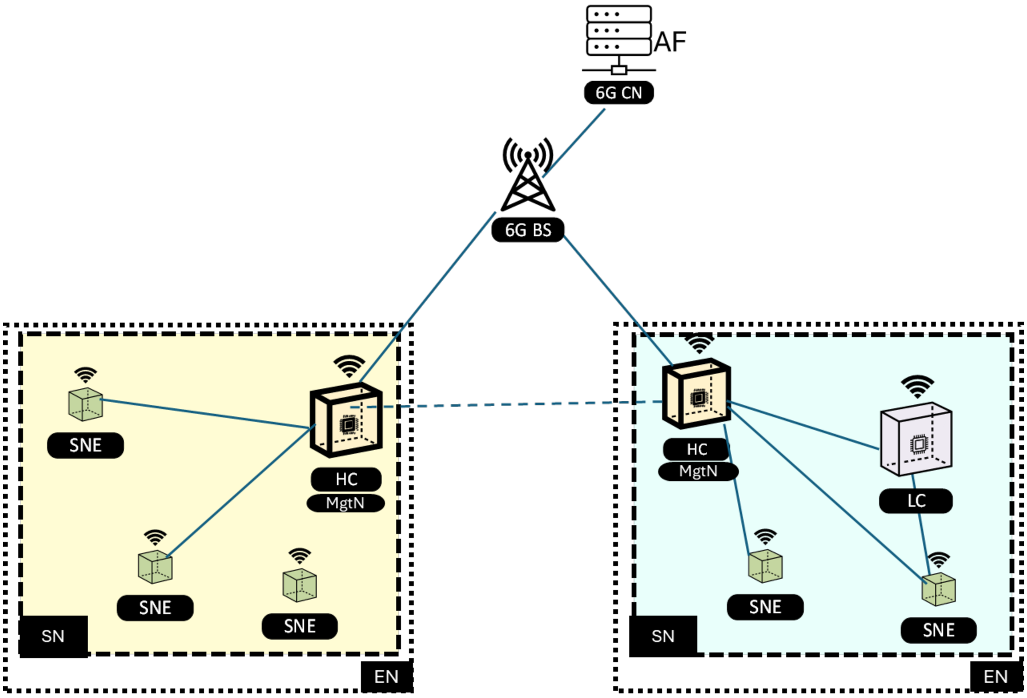


Figure 43 Connections between the devices in and between the subnetworks

The different scenarios need different solutions based on the origin and the termination of the data, and depending e.g., on if there are one or multiple elements which are originators and send data and if there is one, or multiple elements, which receive the multi-modality data.

3.4.2.4.1 Case 1: Multi-modal data from 6G Parent network to a single element in a Subnetwork

In this case there are two (or more) multi-modality data streams from the application in the network to a device (LC#1), see Figure 44. The LC#1 is part of a subnetwork and the data to the device is relayed by the HC device (e.g., HC with management functionality) of the subnetwork.

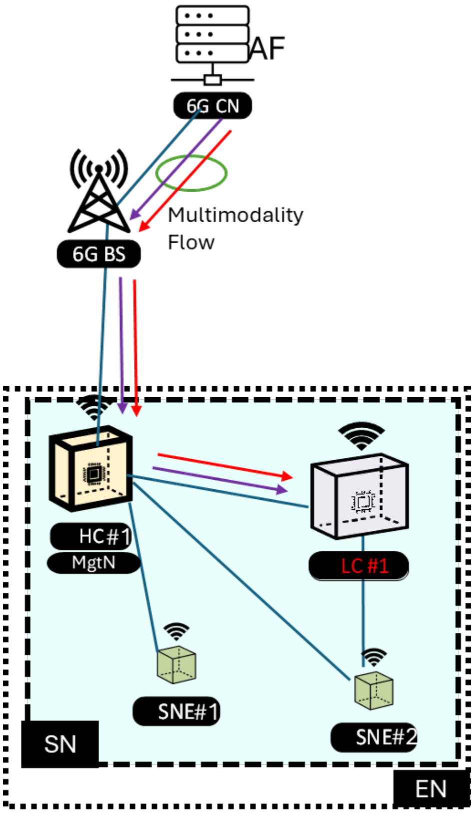


Figure 44 Multi-modal streams from the application to an element in the subnetwork

In this case the multi-modality streams are interrelated, meaning that there is a requirement of the latency of the packets in one flow to be approximately the same as the latency of the packets in the other flow. This means that two packets in the corresponding two data streams sent simultaneously shall be reaching the application in the device at approximately the same time, in order to also be used simultaneously by respective application. The data in the two streams may be sent in different QoS flows with different priority. Since the data streams are interrelated through multi-modality of the same application (e.g. audio and video) there should be some functionality to keep the flows synchronized, the packets need to be handled together in the element with management function (e.g., HC#1) as well as in the 6G BS when scheduled for transmission. In this case 1, there are two interrelated downlink data streams from the application in the network with different QoS requirements where packets that are sent simultaneously in the two streams shall be delivered to the application layer of the element (LC#1) approximately at the same time.

The MgtN (HC#1) and the 6G BS need the information which flows are related based on multi-modality and also which packets in these flows are interrelated and need to be delivered at approximately the same time. This can be integrated into the packet header by the network. Based on that information the packets are handled together so that none of the packets are delayed in excess to the others due to e.g., congestion, see Figure 45.

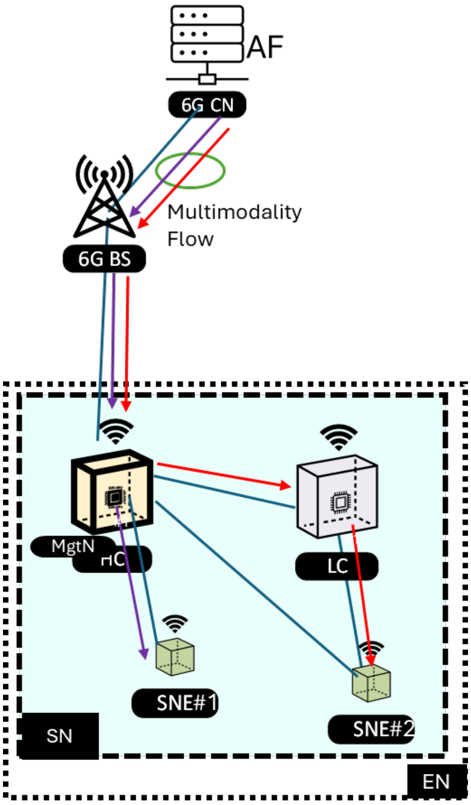


Figure 45 Illustration of applying packet interrelation information to enhance QoS framework

The information which QoS flows are interrelated is on a very basic level known in the 3GPP Core Network in Rel. 18 by sending multimodality information (MMSID) to RAN and to MgtN so the flows can be handled together. This associated handling is done per flow and not per packet. A sequence diagram of the handling of the multimodality for case 1 is shown in Figure 46.

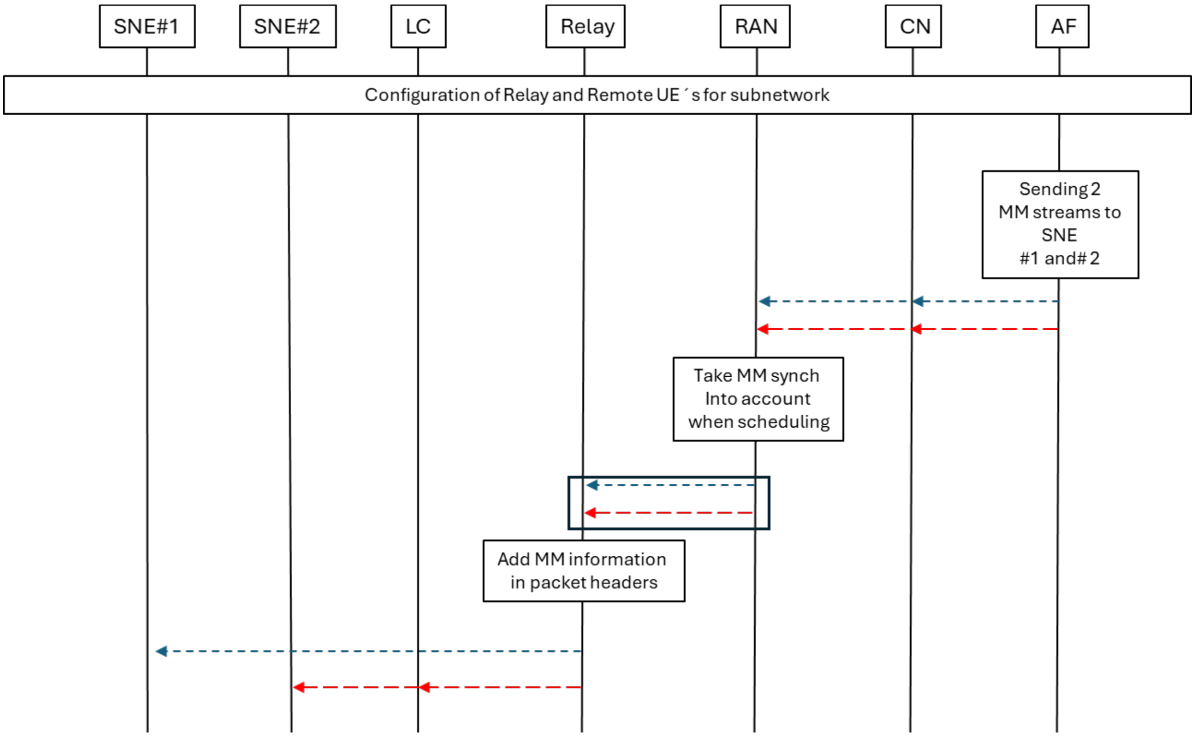
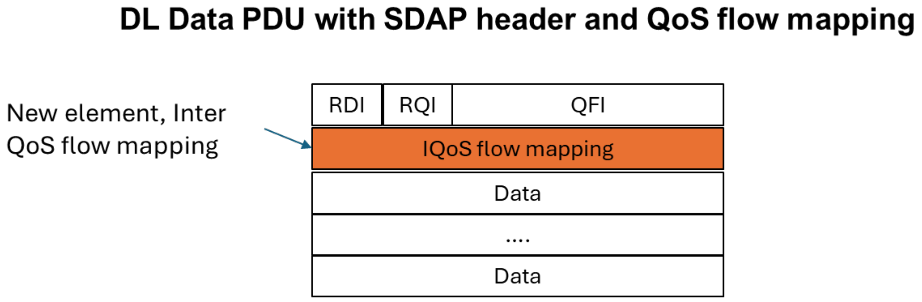

SNE#1
SNE#2
LC
Relay
RAN
CN
AF
Configuration of Relay and Remote UE´s for subnetwork
Sending 2 MM streams to SNE #1 and# 2
Take MM synch Into account when scheduling
Add MM information in packet headers

Figure 46 Sequence diagram for Case 1, when the application in the networks sends multimodality data to one SNE.

When adding the multimodality information in the packet header, it is possible for the lower layer RAN protocols to handle the data packets from the two streams together. To do that, the information needs to include to which other packet(s) a packet is interrelated. This information of interrelated packets could e.g. be added to a packet header which is available to the MgtN, e.g., SRAP or SDAP header (depending on whether the MgtN is a L2 or L3 relay) or in a new protocol defined for 6G. In Figure 47 there is a proposal of an updated Data packet header for the SDAP protocol.



DL Data PDU with SDAP header and QoS flow mapping

New element, Inter QoS flow mapping

RDI	RQI	QFI
IQoS flow mapping		
Data		
....		
Data		

IQoS flow mapping parameters:

- PDB (PDU set or PDU)
- IQPDB (Inter PDUset/PDU packet delay budget)
- Timestamp and/or remaining time.
- QoS flow identifier dependency [QoS id x, QoS id Y]

Figure 47 Enhanced DL Data PDU Header.

This solution proposes adding multimodality synchronization information in the packet headers.

3.4.2.4.2 Case 2: Multimodal data from 6G Parent network to multiple elements in a Subnetwork

In this case, see Figure 48, there are two (or more) multimodality downlink data streams from the application in the network to two (or more) different elements (SNE#1 and #2).

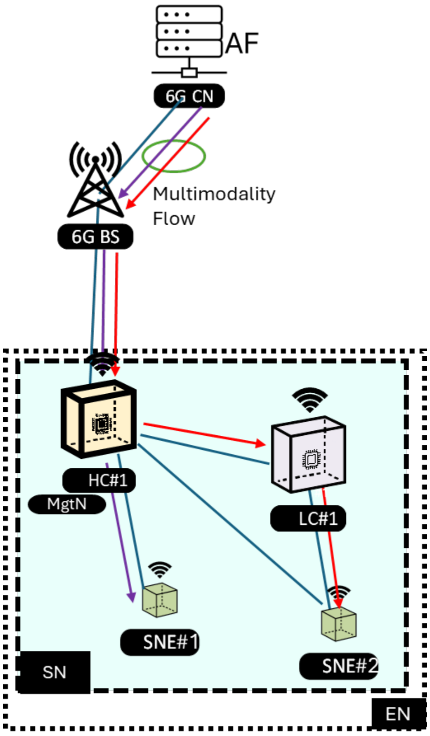
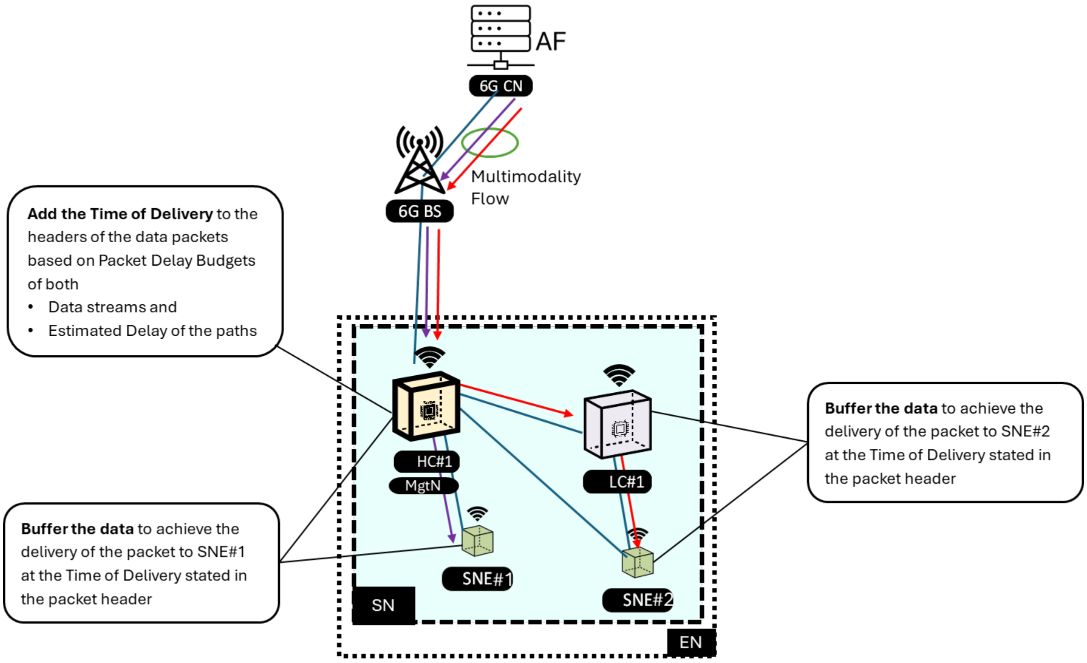


Figure 48 Multimodal streams from the application to several elements in the subnetwork

This case could be an interactive gaming use case where the video related to an occasion is shown at the same time as the corresponding audio, or haptic occasion happens. One of the elements (SNE#2) is connected to with Management function (HC#1) via an extra relay (LC#1), therefore the latency of the two paths from HC#1 to the two elements, SNE1 and SNE2 differs. This makes the synchronization of the delivery of the data to the two elements more difficult since it is not known exactly when the data is available to be delivered to e.g. the application layer in SNEs as shown in Figure 48 above.

In this Case 2, there are two downlink data streams with different QoS requirements which shall be delivered from the application in the NW to the application layer of two different elements simultaneously.

An illustration of this case is shown in Figure 49. In order to deliver the packets simultaneously the deliveries of data need to be synchronized between the paths. One way to do that is to define the delivery time, based on the subnetwork time synchronization. The subnetwork time synchronization is used in the subnetwork to define the structure of the radio interface. All elements in the subnetwork are therefore synchronized.



Add the Time of Delivery to the headers of the data packets based on Packet Delay Budgets of both

- Data streams and
- Estimated Delay of the paths

Buffer the data to achieve the delivery of the packet to SNE#2 at the Time of Delivery stated in the packet header

Buffer the data to achieve the delivery of the packet to SNE#1 at the Time of Delivery stated in the packet header

Figure 49 Illustration of the multimodal streams from the application function in the network to the SNEs #1 andSNE #2

The MgtN (HC#1) in Figure 49 adds a time (delivery time) when the data shall be delivered to the application layer in the UE (in the subnetwork synchronization time base) to a header of the data packet, e.g. as in Figure 47. Based on this time in the header the MgtN (HC#1) or Relay (LC#1) which delivers the packet to the respective UE can buffer the data until it is time to deliver it to the UE, or alternatively the data can be buffered in the modem of the element until it is time to deliver the data packet to the application layer. The time when the data packet shall be delivered according to this procedure should be based on the packet delay budget related to the QoS requirement and on the actual delay to when the packet is available to be delivered, i.e., there is a delay requirement, but also circumstances about the actual delay. This delay needs to be known by the MgtN and could be reported from the respective element in a measurement report to the MgtN.

A sequence diagram of this proposal is shown in Figure 50.

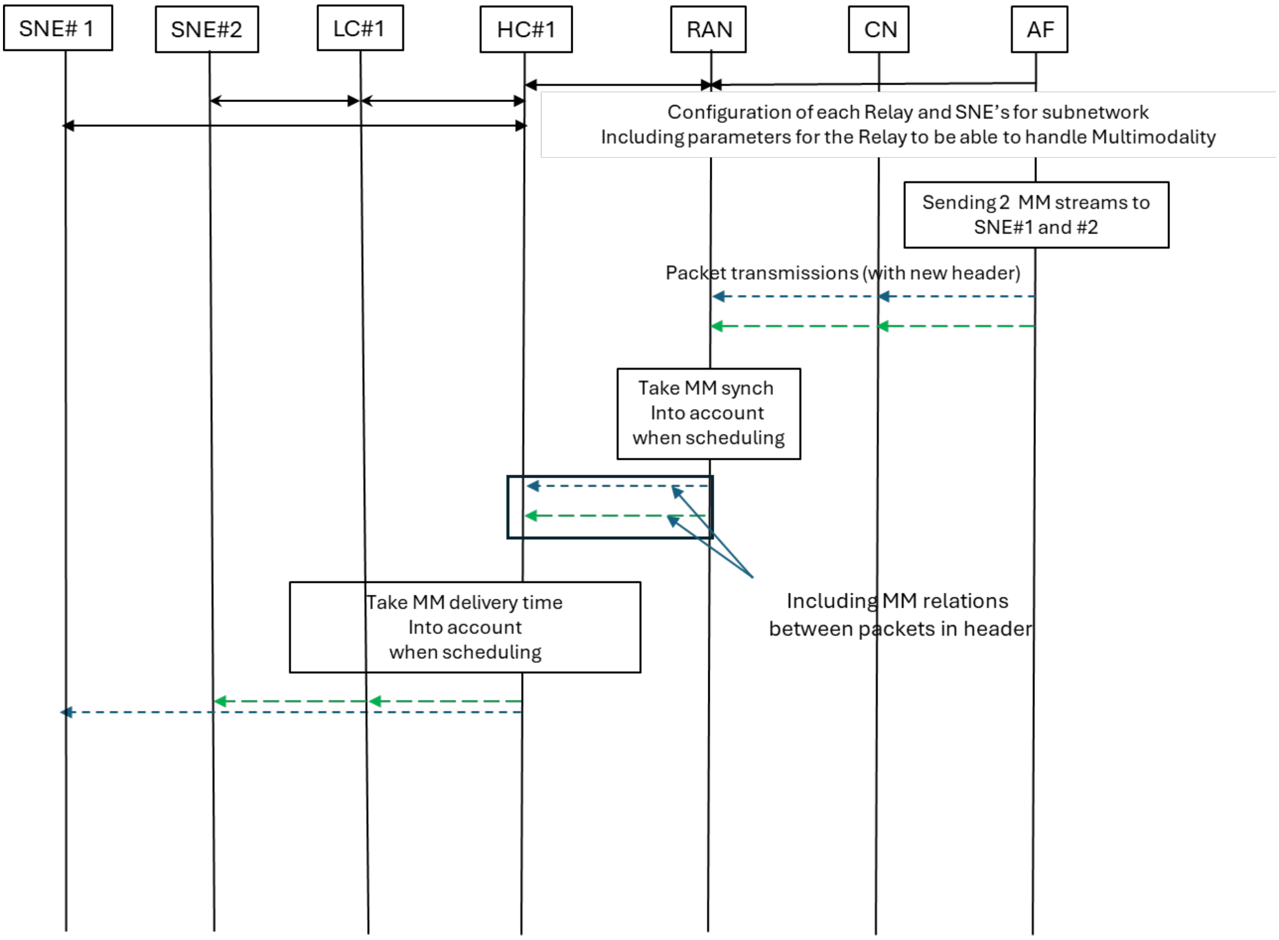

SNE# 1
SNE#2
LC#1
HC#1
RAN
CN
AF
Configuration of each Relay and SNE's for subnetwork
Including parameters for the Relay to be able to handle Multimodality
Sending 2 MM streams to SNE#1 and #2
Packet transmissions (with new header)
Take MM synch Into account when scheduling
Including MM relations between packets in header
Take MM delivery time Into account when scheduling

Figure 50 Sequence diagram for Case 2, when the application in the networks sends multimodality data to multiple SNEs.

In this solution for case 2 it is proposed to add multimodality synchronization information in packet headers including a delivery time to the application layer in the respective device. Thereby it makes it possible for the network to deliver the packets in the two flows to multiple SNEs approximately simultaneous.

3.4.2.4.3 Case 3, Multimodal data from elements in a Subnetwork to a 6G Parent network

In this Case 3, illustrated in Figure 51, the multimodality streams are sent in the uplink from two different elements (SNE#1 and SNE#2). A video game may for example involve several elements, e.g., one for the display, one for the audio and one for the haptic information, these flows are interrelated and therefore form a multimodality use-case. When there is an occasion in the game which affects both video, audio and/or haptic, the different information transfers for this occasion need to be synchronized by the HC with MgtN function in order to deliver them to the application (AF) in the network simultaneously.

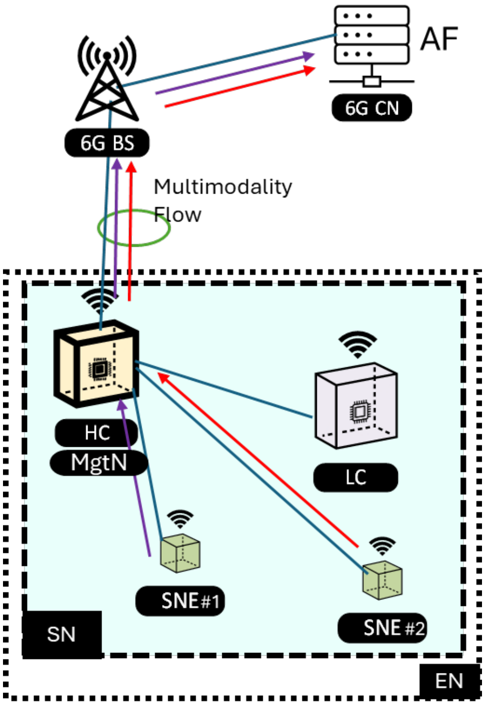


Figure 51 Illustration of Multimodal streams from subnetwork elements to the application function, AF, in the network

The data packets in the two uplink streams in Figure 51 are interrelated, especially the data packets generated simultaneously in the two SNEs need to be delivered together. Therefore, the scheduling from the elements to the MgtN and from the MgtN to the gNB need to take this interrelation into account and handle the interrelated packets together with the goal to deliver them to the application (AF) simultaneously, taking the QoS requirements on delay budget into account.

In case3, the data streams are generated in two different SNEs but are still interrelated based on that several different elements are used within the same use case, e.g. related to the same game application. The problem is how they shall be synchronized and to identify that the packets from different SNEs are interrelated an form a multimodality session. One solution to this problem, as shown in Figure 52 is that the interrelation is added to a packet header in the MgtN. This interrelation can be based on the time when the packets were generated. Thereby the packets in the two streams generated simultaneously will be handled together when scheduling data in the UL in the MgtN to the gNB.

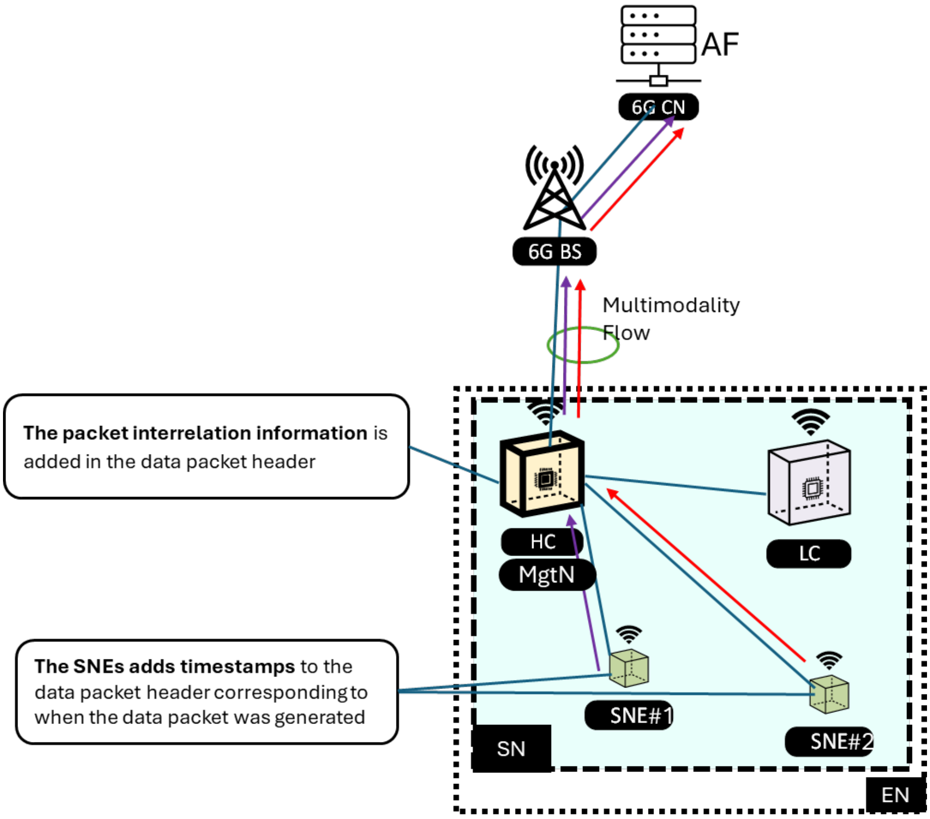

AF
6G CN
6G BS
Multimodality Flow
The packet interrelation information is added in the data packet header
HC
MgtN
LC
The SNEs adds timestamps to the data packet header corresponding to when the data packet was generated
SNE#1
SNE#2
SN
EN

Figure 52 Illustration of the multimodal streams from the multiple elements to the AF in the network

The data packets in streams in Figure 52 are included in a multimodality session and are therefore provided with time stamps by the respectively elements SNE#1 and SNE#2 corresponding to the time, when the packet is generated. This timestamp can be based on the subnetwork timebase and thereby used in the MgtN when deciding which packets that are tightly interrelated and adding this information in the packet header.

When using these timestamps in the HC#1 it is possible to synchronize the data streams in the uplink originating from the different elements. The sequence diagram of this solution is shown Figure 53.

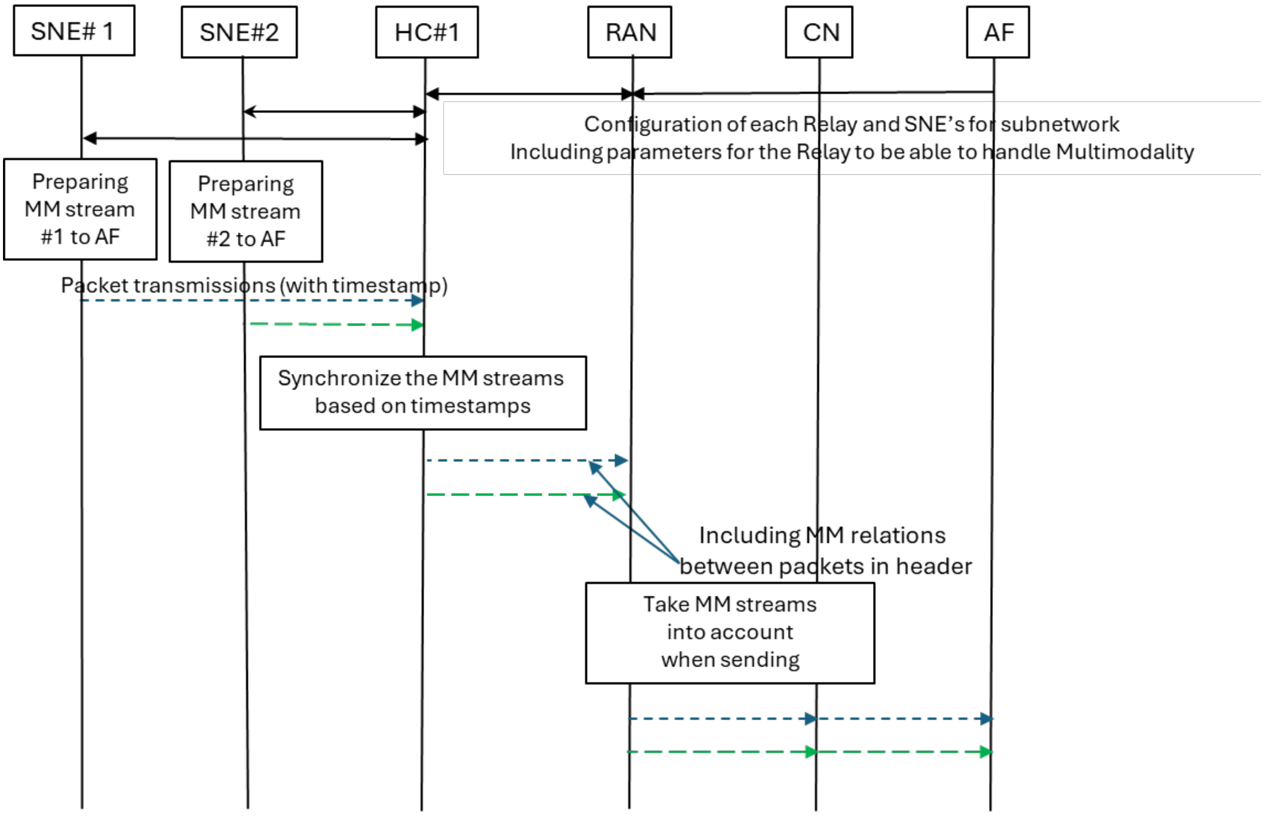

SNE# 1
SNE#2
HC#1
RAN
CN
AF
Configuration of each Relay and SNE's for subnetwork
Including parameters for the Relay to be able to handle Multimodality
Preparing MM stream #1 to AF
Preparing MM stream #2 to AF
Packet transmissions (with timestamp)
Synchronize the MM streams based on timestamps
Including MM relations between packets in header
Take MM streams into account when sending

Figure 53 Sequence diagram for Case 3, when two SNEs sends interrelated multimodality data to the application in the network.

This solution to Case 3 proposes to add Multimodality synchronization information in packet headers from the MgtN (HC#1) to the network including the time when the packets were generated, (received from the SNEs). Thereby it makes it possible for the network to deliver the packets in the two flows to the application in the NW approximately simultaneous.

3.4.2.4.4 Case 4, Multimodal data traffic between elements in different subnetworks

In this scenario, Figure 54, there are two elements SNE#1 and SNE#2, in subnetwork SN#1, transmitting data to other elements SNE#4 and SNE#5, in SN#2, where the dataflows are related to each other, e.g. there is an activity in the first subnetwork where audio and video is sent from different SNEs, one with a microphone and one with a camera, which are sent to the elements in the other subnetwork as a multimodality flow and shall reach the application layer in the target SNEs, SNE#4 and SNE#5, simultaneously in order for the video and audio to be synchronized when played with video on the screen and audio from the loudspeaker.

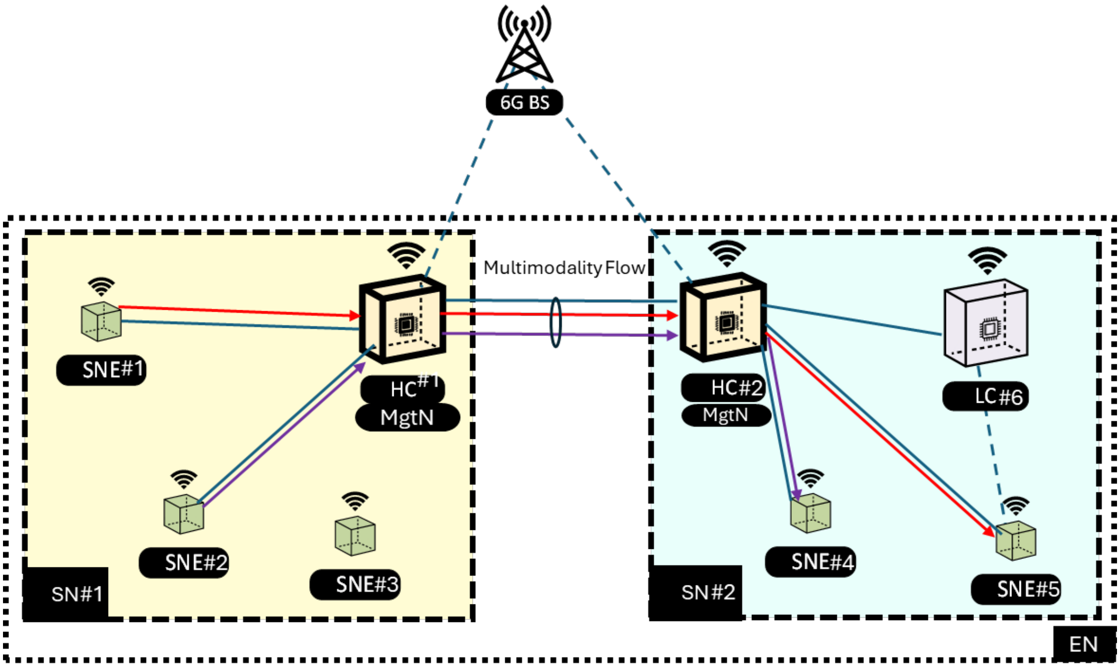


Figure 54 Illustration of case 4: Two elements in SN#1 triggers an action in SN#2

The Multimodal data traffic in Figure 54, between elements in the two subnetworks, which shall be synchronized, are sent from the SNE#1 and SNE#2 in SN#1 to Elements SNE#4 and SNE#5 respectively as multimodal signals. The HC with management function in the first subnetwork detects that the packets from Element SNE#1 and SNE#2 are interrelated so that they shall be received in SNE#4 and SNE#5 respectively, in the order they are generated in the elements. Thereafter the HC with management function in SN#2 shall make sure the multimodal data traffic, are delivered in Elements #4 and #5 in correct time, also fulfilling the delay budget of the QoS requirements of both flows.

This use case describes an element to elements multimodal case. In order to guarantee that the multimodal transmission works the transmitting elements SNE#1 and SNE#2 add timestamps to the packet headers, similar as in the previous section. This is illustrated in Figure 55.

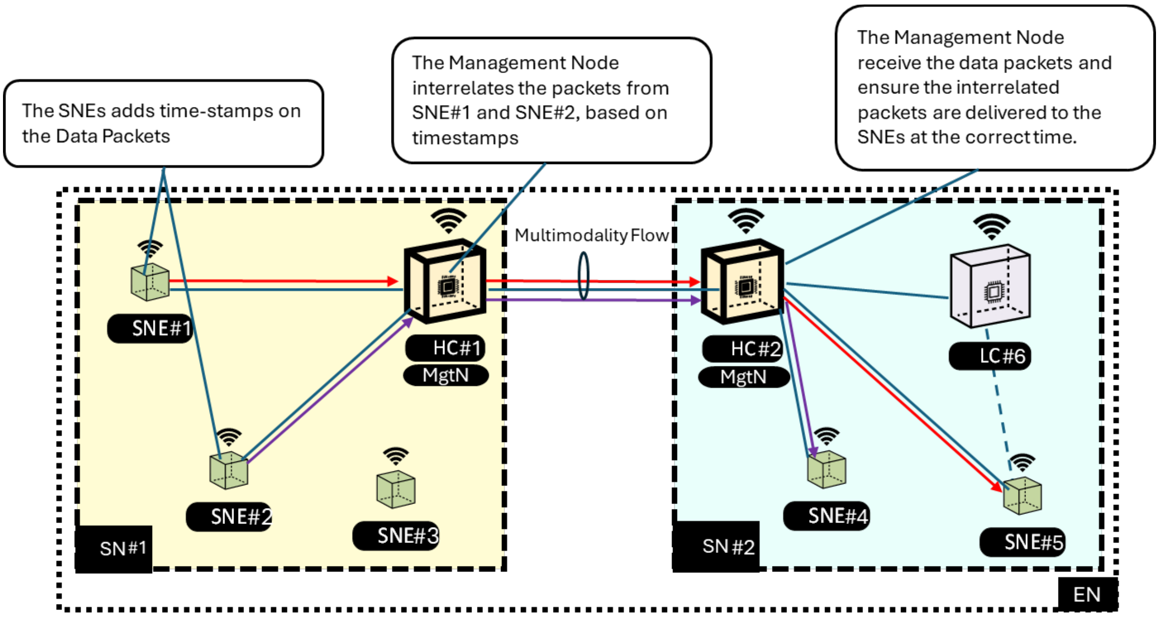

The SNEs adds time-stamps on the Data Packets
The Management Node interrelates the packets from SNE#1 and SNE#2, based on timestamps
The Management Node receive the data packets and ensure the interrelated packets are delivered to the SNEs at the correct time.
Multimodality Flow
SNE#1
HC#1
MgtN
SNE#2
SNE#3
SN#1
HC#2
MgtN
LC#6
SNE#4
SNE#5
SN#2
EN

Figure 55 Illustration of the multimodal streams from the multiple elements in one subnetwork to multiple elements in another subnetwork.

In this solution the MgtN of the first subnetwork can relate the packets transmitted from the two elements. These elements are r transmitted to the MgtN of the second subnetwork. To guarantee a synchronized delivery of the packets this MgtN adds an expected time of delivery to the packets of both data streams. In this case the solutions in case 2 and case 3 above are combined. The data may be buffered either in the MgtN or in the elements SNE#4 and #5 to guarantee simultaneous delivery of the packets to the application layer in the respective element. The sequence diagram of this solution is shown in Figure 56

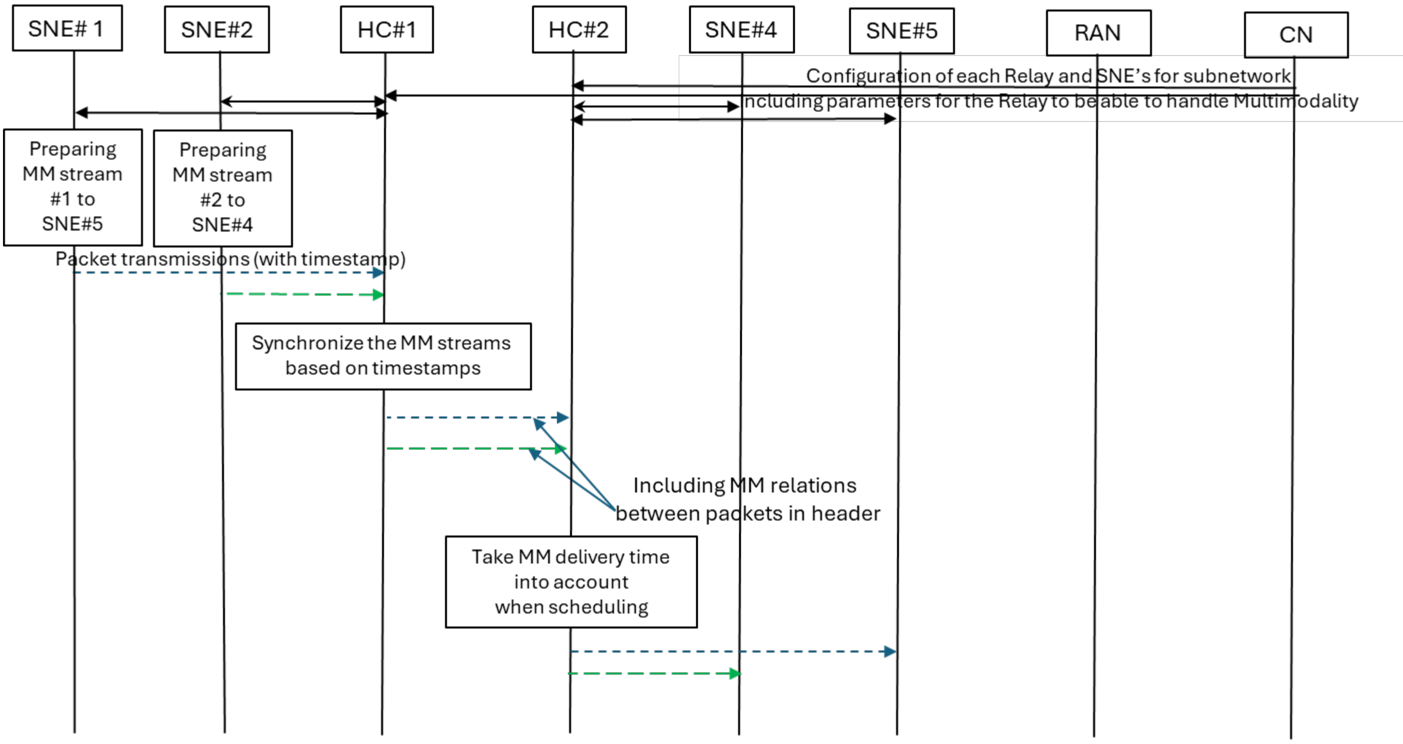

SNE# 1
SNE#2
HC#1
HC#2
SNE#4
SNE#5
RAN
CN
Configuration of each Relay and SNE's for subnetwork
including parameters for the Relay to be able to handle Multimodality
Preparing MM stream #1 to SNE#5
Preparing MM stream #2 to SNE#4
Packet transmissions (with timestamp)
Synchronize the MM streams based on timestamps
Including MM relations between packets in header
Take MM delivery time into account when scheduling

Figure 56 Sequence diagram for Case 4, when two SNEs sends interrelated multimodality data to two other SNEs in another subnetwork.

This solution to case 4 combines the mechanisms of cases 2 and 3. When multiple SNEs transmit the multimodality data and they are sent simultaneously to multiple target SNEs. Thereby it becomes possible for the network to deliver the packets of the two flows to the application on the target SNEs approximately simultaneously.

3.4.3 Study on Subnetwork Scheduling

As highlighted in D4.2, Section 2.2.1.1, the architectural option of having the SN not fully transparent to the 6G overlay NW is advantageous to limiting the required complexity within the HC device acting as MgtN [4]. In particular, the 6G BS shall be made aware of the devices that are associated with a specific MgtN. In this way, only a single physical connection is required for the MgtN and virtual connections to the devices within the SN, as shown in Figure 57. To avoid further complexity in terms of scheduling, as imposed by Integrated Access Backhaul (IAB) [10], where the IAB-nodes are wireless connected BSs, the following proposals aim to keep the scheduling decision at BS side and allow for a leaner MgtN.

The design goal is for the BS to schedule all UEs it has to serve in the same way, regardless of whether they are directly connected via the Uu-Interface, like UE5 in Figure 57, or indirectly connected via the MgtN and residing in a SN, like UE2, UE3 and UE4 in Figure 57. In other words, the BS shall schedule e.g.

UE2 in UL, by sending UE dedicated UL Grant assigned to UE2 to the MgtN (similar to sending DCI with UL Grant to UE5). This solution allows for end-to-end QoS through the SN, controlled by the BS, although the SN might not operate on BS-owned resources.

To enable this scheme, a new per-UE Buffer Status Report (BSR) needs to be introduced, as shown in step (1) of Figure 57, that allows the MgtN to report individual UE buffer status levels and by that enable the BS to perform individual UE scheduling through the SN. The following subsections describe different variants of UE dedicated UL Grant handling in detail, how it can be performed and what different implications to the 6G NW derive from that besides the aforementioned per-UE-BSR.

3.4.3.1 UE Data buffered at the MgtN

As mentioned above, this scheme requires a new per-UE-BSR reporting towards the BS as shown in step (1) of Figure 57. The MgtN has the Gateway (GW) functionality and shall collect per-UE information on the buffer status and report this to the BS. The message sequence chart in Figure 58 shows two alternatives. In the first alternative denoted as “Early BSR” in Figure 58, the MgtN may reuse the mechanism of an early (pre-emptive) BSR to collect information from UEs before that actual data arrives at the MgtN [14]. In the second alternative denoted as “BSR” in Figure 58, the MgtN compiles the per-UE-BSR based on the already buffered UL data it has received from a UE.

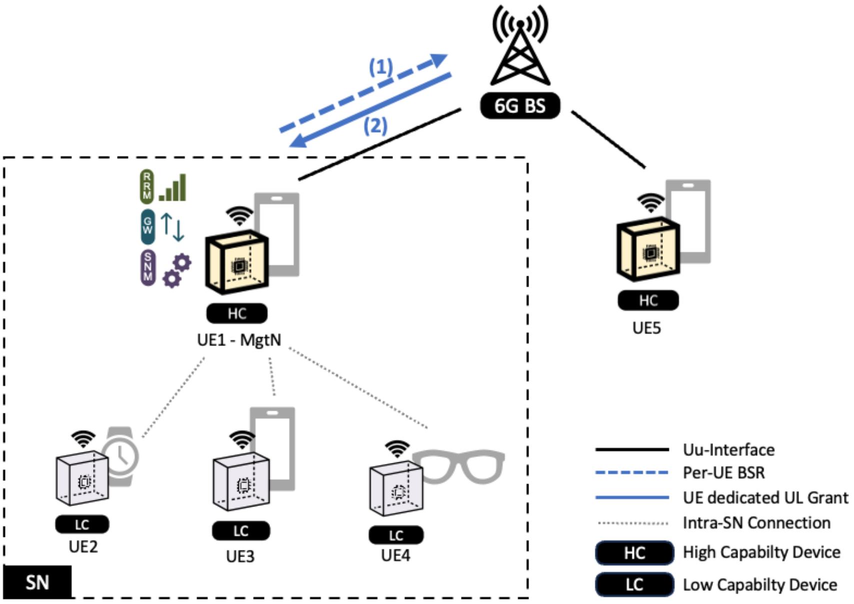


Figure 57 UE dedicated UL Grant with buffered UE Data

Figure 58 Sequence chart of UE dedicated UL Grant with buffered UE Data

Subsequently, the BS scheduler is aware of the individual buffer status for each UE, whether inside or outside a SN and can assign UL grants for individual UEs. To schedule UE2 the BS shall send UL Grant dedicated to UE2 to the MgtN as shown in Step (2) of Figure 57, e.g. via special DCI information or encoded in a MAC CE. In response to that, the MgtN shall provide a Transport Block (TB) to the BS, which contains only UE2 UL data from its buffered UE2 data. Beside the new IEs for Steps (1) and (2), no further enhancements of the Uu-Interface are required. Nevertheless, this solution requires more memory at the MgtN, since the MgtN needs to buffer UL data from all devices in the SN. This approach also adds latency to the UL data due to the additional buffering at the MgtN even though the early BSR scheme of Alt2 in Figure 58 can be used to reduce this delay.

3.4.3.2 UE Data scheduled on demand by the MgtN

Latency is a very important KPI for the use-cases identified D2.2 [2] and since the solution described in the previous subsection increases the UL latency, a more advanced scheme is proposed. In fact, in the new scheme no additional buffering is required. Instead, the MgtN pulls the UL data from a device within the SN on demand precisely when the corresponding UE dedicated UL grant is received from the BS. This requires enhancements on the Uu-Interface, especially in UL to allow for more flexible, more dynamic as well as more relaxed UL Grant handling compared to the legacy in NR. In NR, the N2 parameter describes the capability of a UE on PUSCH preparation time, which in turn derives the BS-configured K2 parameter [11]. The latter is defined as the slot offset between the reception of the DCI for UL scheduling and the reception of the UL [11].

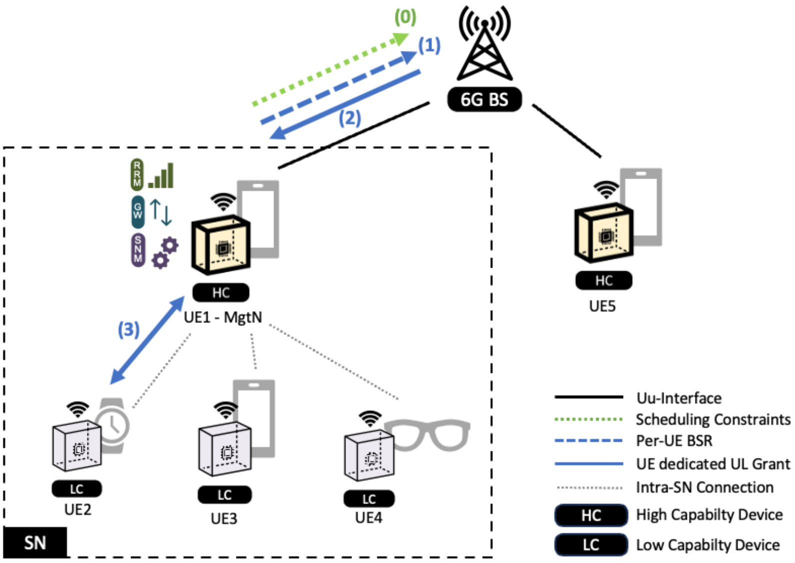


Figure 59 UE dedicated UL Grant with scheduled UE Data

A new MgtN-associated N2 capability per UE is required. Explicitly, this capability should be evaluated by the MgtN to accommodate for the local link conditions, the path to a certain UE and the time MgtN needs to pull the data from that device within the SN. For example, if the intra-SN link is bandwidth limited, or has low quality, the MgtN may then counter for multiple SN-internal transmissions or potential ReTx attempts to derive a UE-specific PUSCH preparation time. The new per-UE-information needs to be reported to the BS, as shown in step (0) of Figure 59 and in the message sequence chart of Figure 60, in order to enable the BS scheduler to consider different PUSCH preparation times for different UEs within the SN. In the context of SNs, different sets of per-UE PUSCH preparation capabilities might be signalled to the BS. For instance, this includes a standalone PUSCH preparation time, whenever the UE is directly connected to the BS (as in legacy 5G), as well as the aforementioned MgtN-associated PUSCH preparation time, when the UE resides within the SN. The latter might be updated periodically or in an event-driven manner based on changing conditions within the SN.

The PUSCH preparation time is only one aspect that requires enhancements to enable on-demand scheduling. Additionally, new per-UE limitations might be signalled to the BS to allow predictable SN scheduling, such as:

- Limited peak throughput of a UE, e.g. by setting a maximum TB size
- Scheduling restrictions to cover SN characteristics and reduce MgtN complexity, e.g.: by defining individual TDD-like patterns for UEs, or by limiting the number of parallel scheduled UEs

Figure 59 describes the message flow for a UE2 being scheduled through the SN. It highlights the newly introduced scheduling constraints in green, and how the BS considers UE2’s constraints in its scheduling decisions. Furthermore, it shows how the MgtN pulls the UL data from UE2 upon receiving the UL grant and how potential HARQ retransmissions of that UL TB can be handled locally by the MgtN without involving the UE again.

Figure 60 Sequence chart of UE dedicated UL Grant with scheduled UE Data

3.4.4 Study on QoS for multi-modality within the Subnetwork

There are many use cases where multiple devices need to work together in a coordinated fashion as a Device Group (DG). In such a setup, each device may have dedicated and thus specific data to send/receive, where data streams originating from the application servers contain a mix of packets for various purposes. These streams are referred to as Service Data Flows (SDFs) e.g. video imaging, audio, control signalling for actuators, sensor data and are depicted in different colours in Figure 61. The originating application servers could be located in the cloud connected to the 3GPP NW as on the left-hand side of Figure 61, or they could even be located within a SN as on the right-hand side of Figure 61.

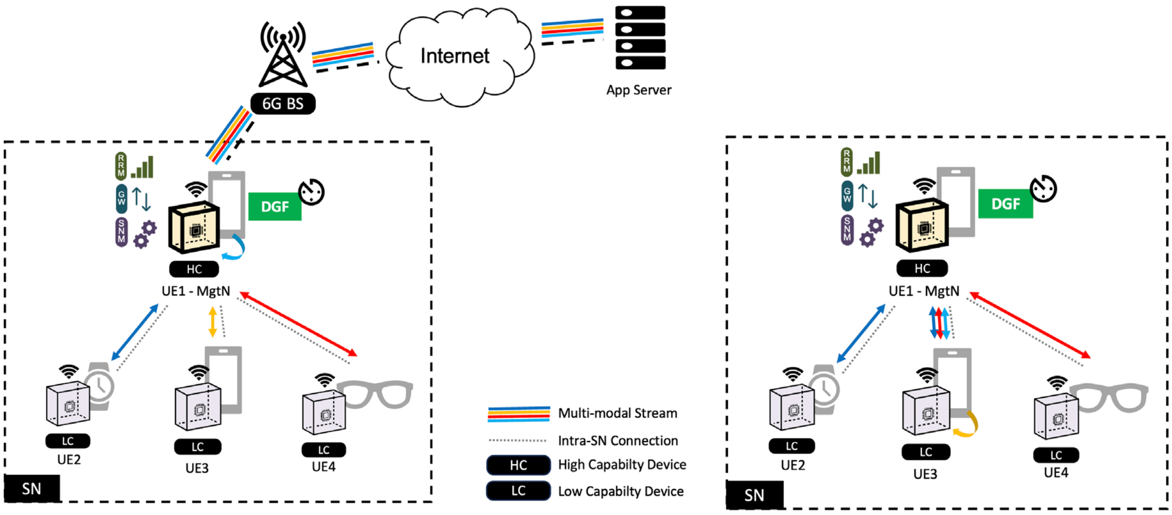


Figure 61 Multi-Modality among Devices in subnetworks

The different SDFs may carry interdependent data. Nevertheless, this data requires to be aligned in time (multi-modal data), while being targeted to the same or to different devices. Note that 3GPP has analysed such multi-modality requirements in [15]. Especially for the AR/VR/XR use cases, which impose tight latency constraints, the associated jitter and synchronicity constraints have been outlined in [1][2]. In these use cases, individual SDFs, which are interdependent, need to be aligned, such as movement with visual feedback and visual data with haptic feedback. Additionally, in multiplayer gaming, where many persons play together, these multi-modality requirements need to be fulfilled, even though everyone has their own individual subscription and hence independent PDU Sessions. In the Industry/Factory use cases involving many robots, sensors and machines, a coordination with each other especially in time is also required. In the use cases of interest [2], the aspect of locality and survivability plays also an important role. Therefore, if a use case is contained in a SN, multi-modal data shall be handled exclusively within the SN without any coordination requirements from the CN. Based on the above, a new QoS Framework that enables inter-device QoS for multi-modal data should be introduced with 6G which will be described in more detail in the upcoming subsections.

3.4.4.1 Coordinated Scheduling of Device Groups

To enable QoS for multi-modal data among multiple and different devices, a new functionality is proposed that resides within the SN, e.g. in the MgtN, which is called Device Group Function (DGF), as presented in Figure 61. The DGF manages an inter-dependent device group (DG) and ensures that DG is served in a coordinated fashion by considering the given packet alignment and data synchronicity requirements. To achieve that, the DGF requires a new metric of Time Alignment on packet/burst/PDU set level. This metric considers propagation delay, link quality and processing latency of individual group members to support scheduling of group members' interdependent SDFs together to achieve synchronicity of multi-modal data at the receiving side. The definition of a group of devices and the DG information itself can be exploited in multiple ways to optimize the SN. For example, the scheduler could use the same slot utilizing MU-MIMO or use SU-MIMO in different sub-bands or in subsequent slots within a certain time window. Another example is UL data scheduling of all DG members. In this case, UL scheduling could be based on SR or BSR of only a single member of the group, which is taking a leading role. This node could inform the MgtN and trigger a scheduling of all DG members.

The independent components of multi-modal SDFs need to be identifiable by the nodes performing the scheduling, such as the overlay 6G BS or the MgtN for the intra-SN scheduling. Traditionally, this is based on filtering rules using e.g. IP addresses, port numbers, next protocol header type, which are applied by UPF or UE to assign certain QFIs. Among multi-modal SDFs aiming at different UEs there are data chunks that need to be synchronized to each other at the receiver side, which is not possible by the current 3GPP framework. Those "SDF chunks" shall be defined on packet level, packet burst level, or PDU set level as shown in below Figure 62.

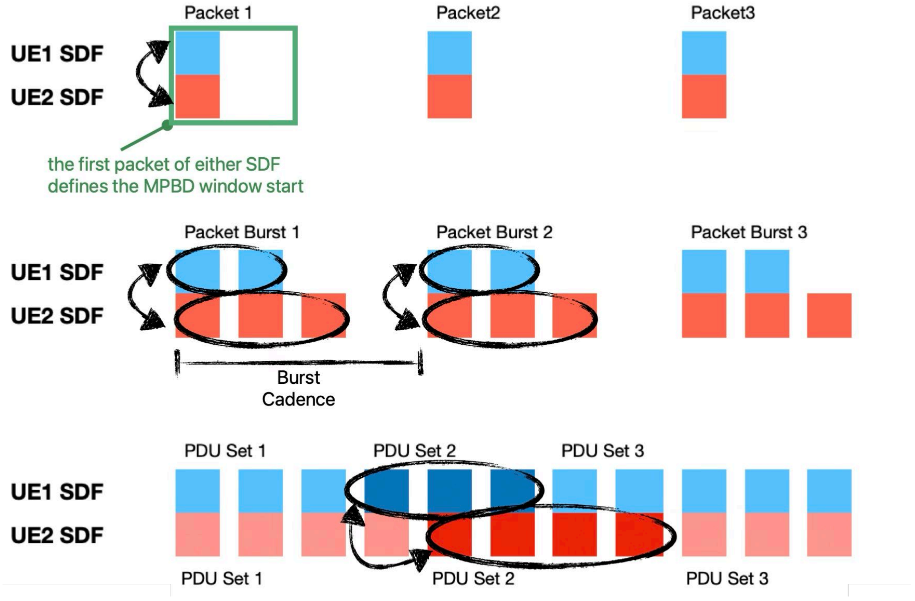


Figure 62 Examples for identifying multi-modality data chunks

Multiple cases are depicted on how data of two different multi-modal SDFs can be mapped.

1:1 Mapping of packets

Packets of different SDFs/QFIs have 1:1 mapping, the first packet to arrive defines the Maximum packet burst distance across devices (MPBD) window. The MPBD defines which packets from different SDFs of different devices belong together and ensures they are timely synchronized and scheduled.

n:m Mapping of packets

SDF chunks of different SDFs/QFIs have fixed/dynamic n:m packet mapping and the identification of the chunks happens based on the timely separation of the individual packet bursts (Burst Cadence). The first packet to arrive of either burst defines the MPBD window start.

SDFs/QFIs PDU set mapping

Within SDFs/QFIs, PDU sets are defined to identify SDF chunks that are subject of the alignment. The time of arrival of the first packet for either PDU set defines the MPBD window start.

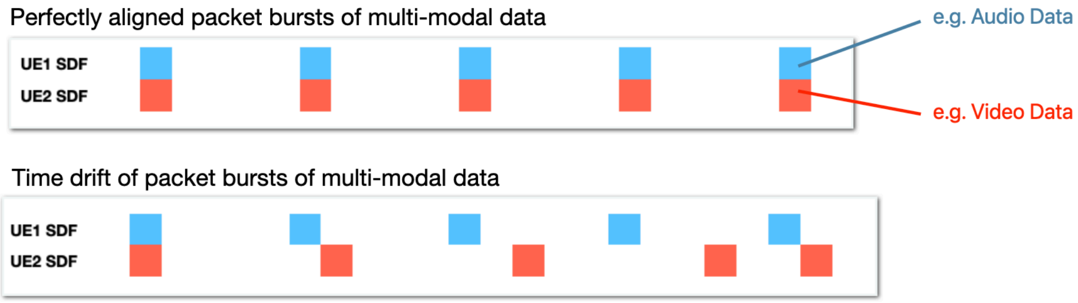
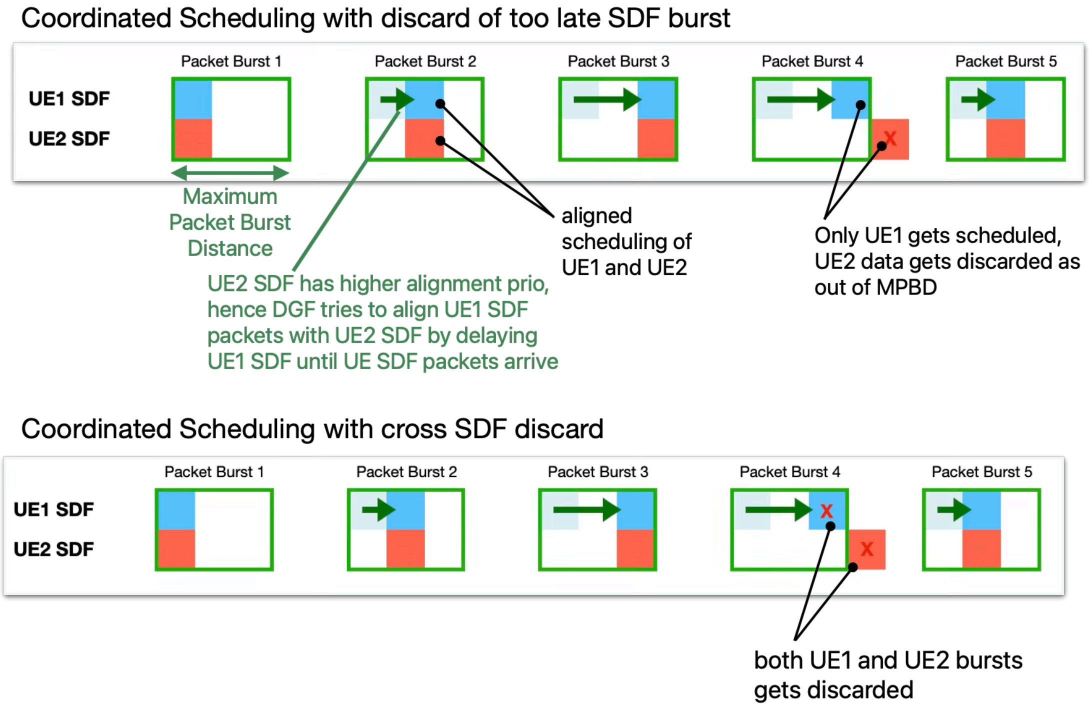

Perfectly aligned packet bursts of multi-modal data
UE1 SDF
UE2 SDF
e.g. Audio Data
e.g. Video Data
Time drift of packet bursts of multi-modal data
UE1 SDF
UE2 SDF

Figure 63 Simplified Scheduling Example

Figure 63 shows an example of perfectly aligned packet bursts versus how the individual bursts may drift away from each other when scheduled independently. To tackle this, the DGF shall enable the scheduler to identify interdependent SDF chunks as mentioned above and to perform a coordinated scheduling of those SDF, as highlighted in Figure 64. The upper part of Figure 64 shows an example of multiple packet bursts arriving over time, and how they are aligned within the MPBD window and scheduled together. In Packet Burst 1 everything is aligned, whereas in Packet Bursts 2, 3 and 5, the data of UE2 SDF arrives late causing the scheduler to delay the data of UE1 SDF to achieve alignment. Packet Burst 4 shows a case, where data arrives outside the MPBD and thus is outdated and must be discarded. In this specific case, the scheduler only delivers the data of UE1 SDF. By contrast, the bottom part of Figure 64 shows an alternative discard strategy. In this figure, the absence of UE2 SDF data within the MPBD window causes the scheduler to even discard UE1 SDF data, in case the delivery of that chunk is not meaningful without delivering its counterpart to UE2.



Coordinated Scheduling with discard of too late SDF burst

Maximum Packet Burst Distance

aligned scheduling of UE1 and UE2

Only UE1 gets scheduled, UE2 data gets discarded as out of MPBD

UE2 SDF has higher alignment prio, hence DGF tries to align UE1 SDF packets with UE2 SDF by delaying UE1 SDF until UE SDF packets arrive

Coordinated Scheduling with cross SDF discard

both UE1 and UE2 bursts gets discarded

Figure 64 Example on aligning packet bursts with coordinated scheduling

Alignment of packet bursts does not necessarily involve scheduling at the same time at the TX side, instead the goal is to achieve alignment on the RX side. Therefore, the coordinated scheduling shall consider Individual Propagation/Processing Delay (IPD) at the receiver to achieve alignment. For instance, the receiver might be an LC device with less processing capabilities and thus may require earlier scheduling to achieve the "playback" of multi-modal data at the same point in time at the receiver side as in Figure 65.

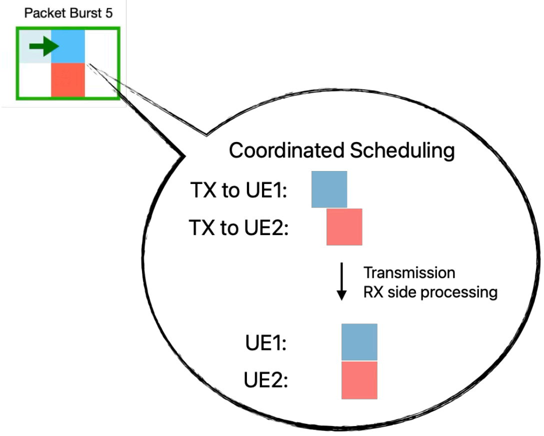


Figure 65 Example for coordinated scheduling to achieve alignment on the RX side

To enable such coordinated scheduling supported by a DGF, the MgtN shall support new procedures to collect new parameters from the group devices within the SN.

Parameters:

- QoS parameter(s) for packet/burst alignment and jitter constrains of SDFs across members
  - Maximum packet burst distance across devices (MPBD)
  - SDF priorities among the interdependent SDFs
  - Discard policy if packet bursts of different SDFs do not arrive within MPBD
    - discard only the delayed burst
    - discard also the arrived burst (due to its dependency to the delayed burst)
- Metric of individual propagation/processing delay (IPD) per DG member
- Vicinity and Quasi Co-Location (QCL) information of UEs towards the MgtN
  - Similar to TCI (Transmission Control Indication) States in 5G [11] albeit from MgtN perspective, such as proximity of devices, a common trajectory or speed, or similar channel conditions

In the message sequence charts of Figure 66 - Figure 68, the DGF is shown as part of the MgtN as suggested in Figure 61. In fact, as the MgtN manages the operation of the SN, including the DGF to the MgtN role may be deemed as the most natural deployment of the DGF for SN operation. However, there are multiple deployment options possible. For instance, the DGF might as well be deployed in the BS or even in the core network. Figure 66 shows how UEs perform Group Registration providing the above-mentioned parameters towards the DGF. In Figure 67, the process of how the DGF sets up the group is described, where it might provide information towards the UPF to ensure the coordinated scheduling along the path through the CN to the BS. In addition, it uses the group information to aid scheduling and mobility decisions knowing that different group members require aligned scheduling and that they may have some QCL-relationship. As shown in Figure 67 and in more detail in Figure 68, the MgtN hosting the DGF may setup additional measurement and reporting on the DG members to keep its group information up to date.

Figure 66 Group registration to DGF

Figure 67 Group Setup Procedure by MgtN

Figure 68 Group Member Measurements

3.4.5 Summary

In this study we have investigated the benefits of utilizing multimodality information to improve and enhance a synchronized delivery of packets belonging to different QoS flows but within the same multi-modality stream. We have proposed various methods where one method is to add multimodality synchronization information to the packets. By adding packet interdependency information to the packet header, the scheduler can gain knowledge not only of the relation between different flows but also the relation between packets.

Furthermore, a method is proposed to add Multimodality synchronization information in packet headers including a delivery time to the application layer in the respective device. Thereby it makes it possible for the network to deliver the packets in the two flows to multiple SNEs approximately simultaneously. By a variant method for UL traffic, it is proposed to add Multimodality synchronization information in packet headers from the MgtN to the network including the time when the packets were generated, (received from the SNEs). Thereby it makes it possible for the network to deliver the packets in the two flows to the application in the NW approximately simultaneous.

In a subnetwork to subnetwork multi-hop scenario, both time stamp and expected delivery times can be used by the multiple MgtN involved to handle a synchronized delivery between multiple SNEs in one subnetwork, to multiple SNEs in another subnetwork.

To achieve equal scheduling opportunities within the subnetwork, the MgtN is proposed to collect buffer status information from all devices within the network are report jointly to the BS, or order to get the scheduling grants in a synchronized manner.

4 DYNAMIC COMPUTATIONAL RESOURCES OFFLOADING WITHIN SUBNETWORKS, AMONG SUBNETWORKS AND TO 6G EDGE-CLOUD

The increasing demand for computationally intensive, latency-sensitive applications—such as augmented and virtual reality (AR/VR), autonomous systems, and AI-driven services—has exposed the limitations of current network architectures, particularly in their ability to support integrated communication and computation. Traditional frameworks like 3GPP primarily address data transmission and quality of service (QoS) metrics related to communication, without accommodating the growing need for distributed computing support. To overcome these challenges, this chapter explores a comprehensive set of frameworks and mechanisms aimed at enabling joint communication-computation resource management in next-generation networks.

In Section 4.1, a distributed compute framework is proposed to support offloading across heterogeneous network nodes. This framework enables low-capability nodes to offload complex computation to nearby, more powerful devices, enhancing system-wide performance and flexibility. Section 4.2 builds upon this by presenting the Quality of Computation Service (QoCS) framework, a novel extension of traditional QoS paradigms. QoCS enables end-to-end management of both network and computational requirements, ensuring reliable performance for data-intensive applications in mobile networks. In Section 4.3, a deterministic task offloading, and resource allocation strategy is proposed for managing workloads across the IoT-edge-cloud continuum. The focus is on meeting task deadlines rather than minimizing latency alone, allowing for more efficient and balanced resource usage. This approach promotes system-wide efficiency and scalability. Section 4.4 adapts the deterministic scheduling strategy to in-vehicle networks (IVNs), highlighting its ability to support centralized computing and distributed tasks within modern automotive systems. The framework is validated in various IVN configurations, including hybrid setups with wireless connectivity. Finally, Section 4.5 introduces the Compute Aware Traffic Steering (CATS) framework for 6G subnetworks, which integrates compute and network-aware service selection. Through dynamic traffic steering and mobility-aware service anchoring, CATS ensures optimal service delivery even under constrained and mobile conditions.

4.1 PROTOCOLS AND PROCEDURES FOR COMPUTATIONAL OFFLOADING

4.1.1 Introduction

The new use cases described in D2.4 [2] with deployments of nodes with variable capabilities demand coordination of the nodes so that they collectively increase their capabilities. So far in this report, this has been achieved by performing functional offloading, i.e. by distributing network and UE functionality across the SNs. Besides functional offloading, the SNs can also enable general computation offloading enabling the deployment of computationally-heavy applications into nodes of lower capabilities.

The current 3GPP framework [10] focuses exclusively on data routing without considering computational offloading. A converged communication and computation architecture is envisioned in D2.4 [3] for all the use case categories [1], which allows LC devices to harness the SN resources to improve their capabilities. Consequently, a distributed compute framework should be proposed to enable this functionality within the SNs. Following the user-centric SN architecture of D2.4 [3], the necessary procedures should involve the overlay BS to the minimum extent, thus retaining the control of offloading within the SN.

To leverage the proposed SN architecture in D2.4 [3] and to further enhance the available computational resources of LC UEs within a SN, it is foreseen that this procedural framework consists of three stages. As shown in Figure 69, Stage#1 of SN creation involves the messaging exchange that enables the connection establishment of local UE(s) with a MgtN. Stage#2 is the SN registration phase, involving messaging exchange between the MgtN and the BS to register the MgtN and its underlying SN with the NW. Finally, Stage#3 defines the messaging exchange between local UE(s) and the MgtN of the serving SN to enable the offloading of one or more compute task(s) and the reception of the respective compute result(s). The message exchange could either be done only locally within a single SN. This case is referred to as *Local Distributed Compute*. Another possible deployment for this message exchange is between different SNs in a direct manner without any cellular NW involvement, or by utilizing network resources exclusively as a communication backbone. Naturally, in the case where the message exchange takes place between different SN, as the NW has no coordination role or control on the offloading, it is referred to as *Decentralized Distributed Compute*. Additionally, it constitutes an extension to the Local Distributed Compute.

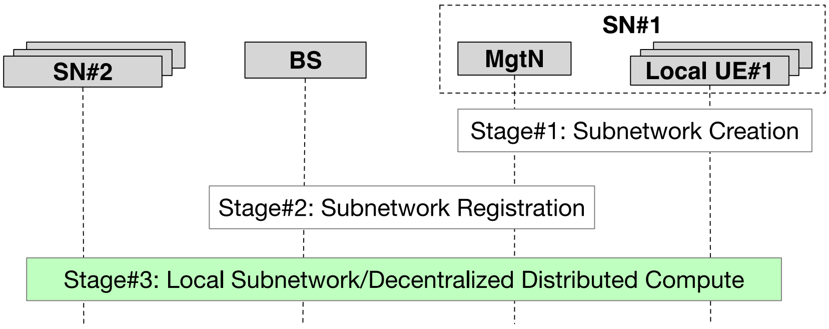

SN#1
SN#2
BS
MgtN
Local UE#1
Stage#1: Subnetwork Creation
Stage#2: Subnetwork Registration
Stage#3: Local Subnetwork/Decentralized Distributed Compute

Figure 69 Proposed three stages procedures for enabling local subnetwork and decentralized distributed compute.

This Section focuses on “Stage#3: Local Subnetwork/Decentralized Distributed Compute”. A set of roles has also been introduced in D2.4 [3] for the sake of enabling local computing as follows:

- **Offloading node (ON):** connected to a SN, having a compute task to be offloaded to one or more Computing Nodes
- **Computing node (CompN):** SN node with certain processing capabilities to perform an offloaded compute task and produce compute result
- **Compute offload controlling node (CCN):** collects all compute capabilities from all available Computing Nodes and makes compute offload decision based on their current load
- **Routing node (RN):** an optional network node at which the compute task/compute result from Offload node/Compute node gets routed to one or more Computing node(s)/Offload Node.

More specifically, to enable decentralized computing, the following roles have also been introduced in [D2.4]:

- **Managing CCN:** a CCN that takes control of the overall distribution logic in addition to the handling of the resource and process management functions
- **Supporting CCN:** a CCN that delegates some or all the compute distribution logic as well as the management of resource and process management functions.

The procedural details for enabling Local as well as Decentralized Distributed Compute will be presented in Sections 4.1.2 and 4.1.3, respectively. Note that the presented procedures are based on the contribution [23].

4.1.2 Local Distributed Compute

In this case, compute tasks are offloaded only to local CompN(s), like the MgtN and/or HC device(s) that are available within the SN, without any NW involvement. Therefore, Stage#2 is optional, since there is no need for registering to the NW, provided that the computation offload is kept within the local SN. The messaging exchange of Stage#3 may either be via the MgtN or via a direct ON to CompN communication. In Stage#3 there are two possibilities in deriving the node that will act as the CCN. The first approach is the MgtN-controlled, where the MgtN is the default CCN and has an active role upon selecting which node will be selected as the CCN. The second approach is a fully decentralized one, where the nodes enter a negotiation phase so that they collectively agree on the node taking the CCN role, which is referred to as *SN CCN*.

As far as the MgtN-controlled approach is concerned, Figure 70 shows a high-level MSC with messaging exchange between the different SN nodes to enable the local SN compute offload procedure. The procedure starts with all CompN(s) in the SN updating the CCN with their compute capabilities, via a "Compute Capabilities Update" message. This message can be sent periodically or event-triggered. Alt#1 "SN CCN Controlled" in Figure 70 constitutes a centralized approach, where the SN CCN aggregates the available compute capabilities and announces them to all the ONs in the SN. By contrast, Alt#2 "ON Controlled" in Figure 70 is a decentralized approach, where no aggregation of compute capabilities takes place at SN CCN, only a simple forwarding of different CompN Capabilities to available ONs. It is left up to the ONs to choose which of the CompN(s) the task(s) would be offloaded to. This alternative provides more flexibility for CompN selection and more privacy, at the expense of a suboptimal load distribution. Once an ON has a computation task to be offloaded, it would inform the CCN via a "Computation Offload Request" message. This request could optionally include a request for a specific CompN to which the ON would like the compute task(s) to be offloaded. The CCN shall evaluate the compute request by the ON and decide whether such request can be fulfilled or not. If the request can be fulfilled, the CCN would in turn send a compute request to one or more CompN(s) and wait for their responses. If the CompN(s) accept(s) the compute offload request, the CCN shall inform the respective ON and the compute offload procedure would commence. Note that the CCN could respond to an ON with an early rejection of the compute request in case it is already aware that such compute task(s) cannot be fulfilled by the available CompN(s) in the SN.

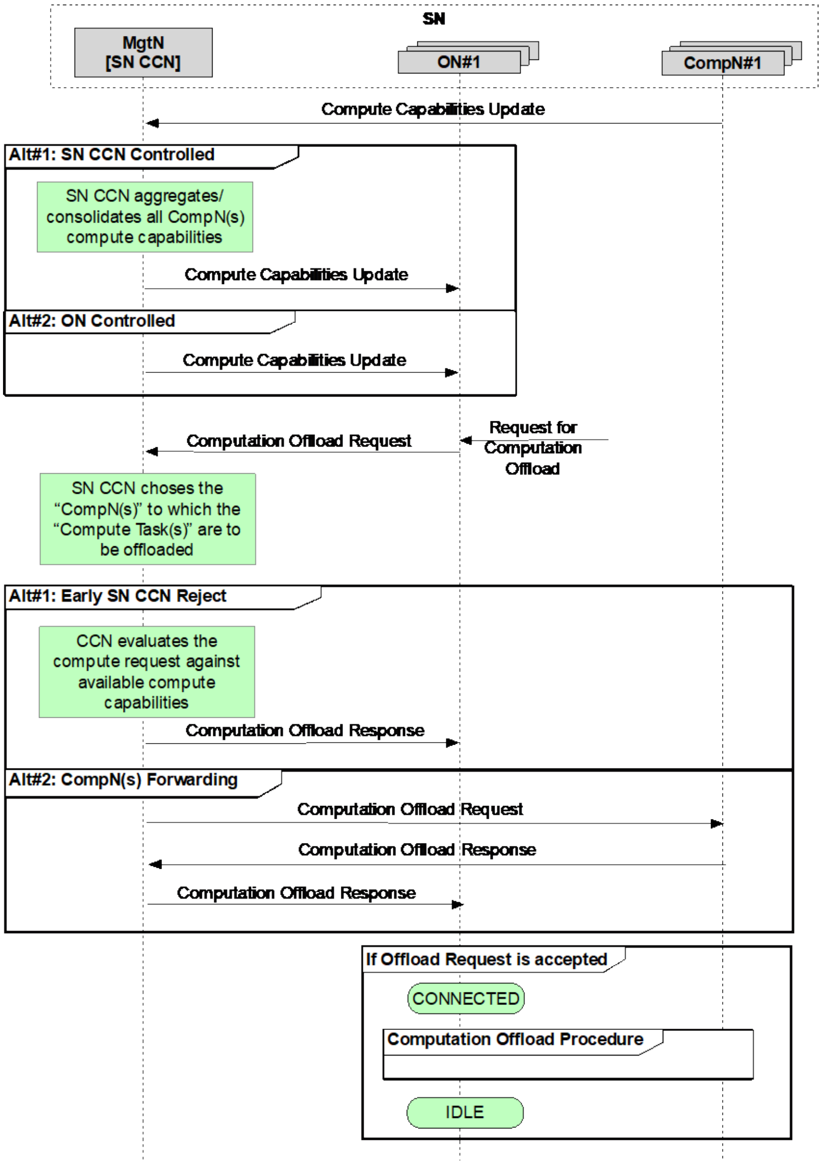

SN
MgtN [SN CCN]
ON#1
CompN#1
Compute Capabilities Update
Alt#1: SN CCN Controlled
SN CCN aggregates/ consolidates all CompN(s) compute capabilities
Compute Capabilities Update
Alt#2: ON Controlled
Compute Capabilities Update
Computation Offload Request
Request for Computation Offload
SN CCN choses the "CompN(s)" to which the "Compute Task(s)" are to be offloaded
Alt#1: Early SN CCN Reject
CCN evaluates the compute request against available compute capabilities
Computation Offload Response
Alt#2: CompN(s) Forwarding
Computation Offload Request
Computation Offload Response
Computation Offload Response
If Offload Request is accepted
CONNECTED
Computation Offload Procedure
IDLE

Figure 70 MgtN-controlled compute offload MSC [23]

Moving on to the fully decentralized approach, Figure 71 shows an MSC for the SN CCN negotiations among the different SN entities. The first option is MgtN controlled, where the MgtN decides which of the available CCNs would act as the SN CCN and sends indication messages accordingly. Note that the indication message from the MgtN could be extended to provide a list of CCNs that could act as the SN CCN (i.e., primary, secondary, etc.), that may be used for fast recovery in case of failure in the primary SN CCN. In the second option, the MgtN has no central role in the SN CCN choice, and it is kept up to the different Nodes to negotiate. The first alternative in Option#2 assumes a broadcast-based approach, where all nodes broadcast their compute capabilities and requests along with their CCN IDs, evaluate the status of the other CCN, and decide on whether to wait to receive or send a "SN CCN Indication". The second alternative in Option#2 assumes a request-based approach, where one UE would send a "CCN Status Request" indicating compute capability and request along with its CCN ID, while the other UE would respond with a "CCN Status Update" indicating its compute capability/request along with a response on whether it accepts or rejects the role of a SN CCN. Finally, in case of CompN taking the SN CCN role, as shown in Option#2, an ON would have to restart the "SN CCN Negotiation" procedure if there is a sudden failure of the chosen SN CCN.

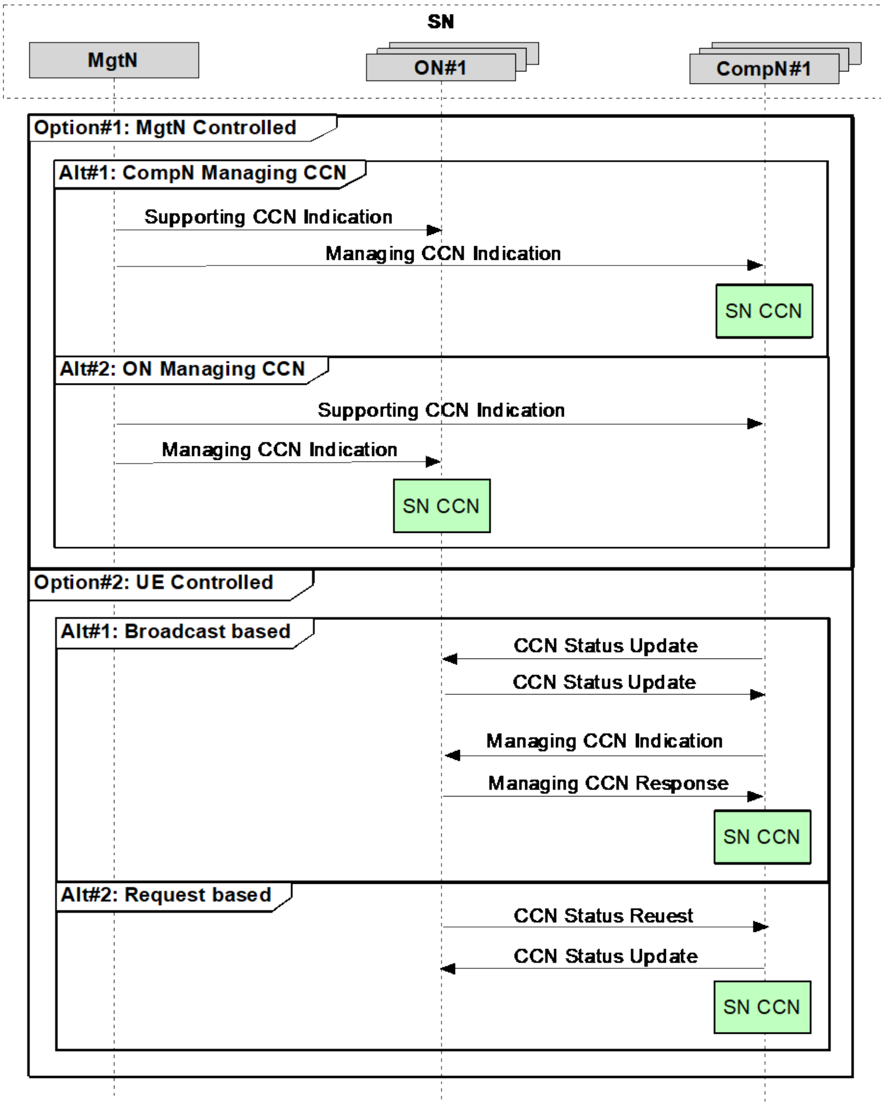


Figure 71 Subnetwork CCN negotiation MSC [23].

After the CCN is selected in the decentralized negotiation approach, an additional step is required to enable ON-CompN direct communication so that the offloading task can be routed from the ON to the CompN. Figure 72 shows a high-level MSC for the messaging exchange between ON(s) and CompN(s) to enable a direct ON-CompN computation offload without MgtN involvement. As highlighted in the figure, during the SN creation phase, some parameters might need to be exchanged within the SN in order to allow direct device to device communication, such as transmit and receive resource pools and discovery resources as described in 3GPP Sidelink [9], or any proprietary pairing information.

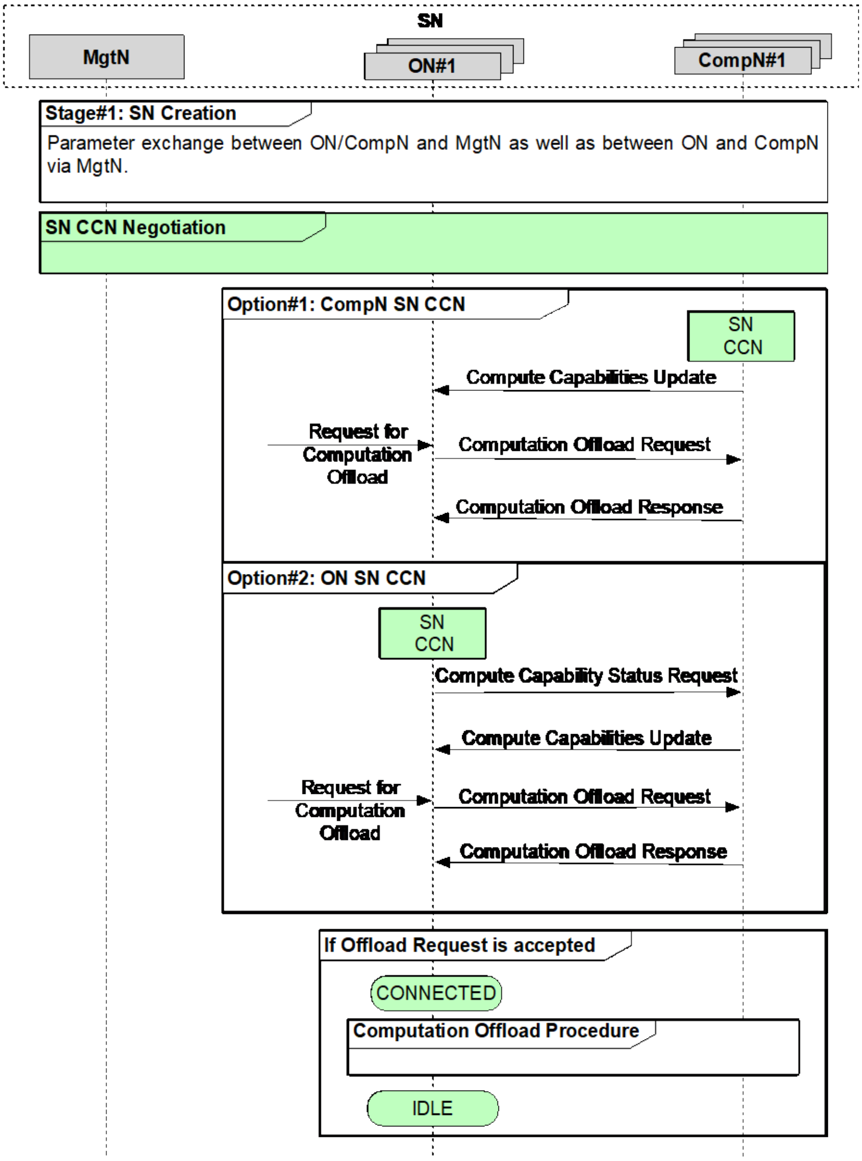

SN
MgtN
ON#1
CompN#1
Stage#1: SN Creation
Parameter exchange between ON/CompN and MgtN as well as between ON and CompN via MgtN.
SN CCN Negotiation
Option#1: CompN SN CCN
SN CCN
Compute Capabilities Update
Request for Computation Offload
Computation Offload Request
Computation Offload Response
Option#2: ON SN CCN
SN CCN
Compute Capability Status Request
Compute Capabilities Update
Request for Computation Offload
Computation Offload Request
Computation Offload Response
If Offload Request is accepted
CONNECTED
Computation Offload Procedure
IDLE

Figure 72 Direct ON-CompN Compute Offload MSC [23].

4.1.3 Decentralized Distributed Compute

In the decentralized distributed compute option, it is assumed that the computation offload is distributed among nodes of different SNs. This option entails the involvement of entities such as the BS and multiple MgtNs of the different SNs. Naturally, negotiations among the different CCNs of the different nodes are needed to decide the role of each CCN in the computation offload procedure. In this specific case, the extended CCN roles will be utilized, namely those of the Managing and Supporting CCN presented in D2.4 [3] as well as at the start of this Section.

In the absence of a central control from the NW, negotiations must take place between the different CCNs for determining which node would act as a managing CCN and which would be a supporting CCN. Figure 73 highlights a proposal for such negotiations, where all available CCNs of different SNs exchange "CCN Status Update" messages, indicating their status, like e.g., battery level, compute capacity within its local SN, their current load or Rx signal strength. This message could be transmitted either periodically or be event-triggered. Based on the NW's or SN's own CCN status, as well as on the received status updates of all other available CCNs, each node would evaluate whether to wait to receive or send a "Managing CCN Indication", the latter indicating that a node would like to take over the role as a managing CCN. Each NW and/or SN CCN entity that receives the "Managing CCN Indication" message decides whether to confirm or reject the indication via a "Managing CCN Response" message. If a CCN has received a "Managing CCN Indication" from another CCN to which it had also sent an indication, it would have to compare its own metrics with those that it has received and then either "confirm" or "reject" based on which CCN is better. Additionally, if a CCN gets a "Managing CCN Indication" from multiple CCNs, it shall decide which one is suited better, "confirm" that one and "reject" all others. Note that if the "Managing CCN" disappears and the need for decentralized compute offload is still there, re-negotiations with all available CCNs would take place to decide on a new "Managing CCN".

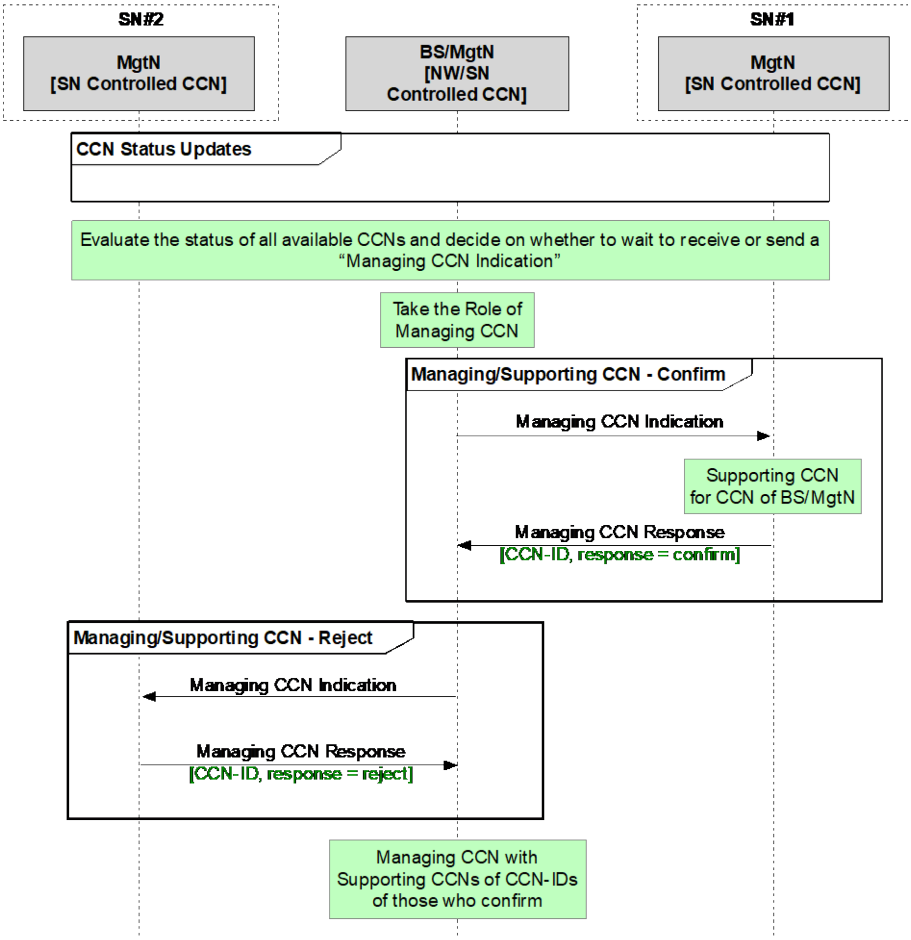

SN#2
MgtN
[SN Controlled CCN]
BS/MgtN
[NW/SN
Controlled CCN]
SN#1
MgtN
[SN Controlled CCN]
CCN Status Updates
Evaluate the status of all available CCNs and decide on whether to wait to receive or send a "Managing CCN Indication"
Take the Role of Managing CCN
Managing/Supporting CCN - Confirm
Managing CCN Indication
Supporting CCN for CCN of BS/MgtN
Managing CCN Response
[CCN-ID, response = confirm]
Managing/Supporting CCN - Reject
Managing CCN Indication
Managing CCN Response
[CCN-ID, response = reject]
Managing CCN with Supporting CCNs of CCN-IDs of those who confirm

Figure 73 Managing and supporting CCN negotiations MSC using a periodic or event- triggered status update [23].

Figure 74 highlights a different approach for realizing the managing-supporting CCN negotiations procedure as request-based approach for exchanging CCN status updates. The "CCN Status Request" message is sent by the "Managing CCN" to the "Support CCNs". This message indicates the node's compute capacity and computation requests, received signal strength among other parameters. This message indicates a node's intention to targeted CCN to act as its "Managing CCN". The recipient of the request shall then respond with a "CCN Status Update" message that includes this node's compute capacity, computation requests, received signal strength among other parameters and also a response message either confirming or rejecting the request. If it confirms, then the responding CCN is now acting as the managing CCN of the requesting CCN node, otherwise the requesting CCN would have to request from another CCN via sending another "CCN Status Request" message, highlighted with the messaging exchange of SN#2 in Figure 74.

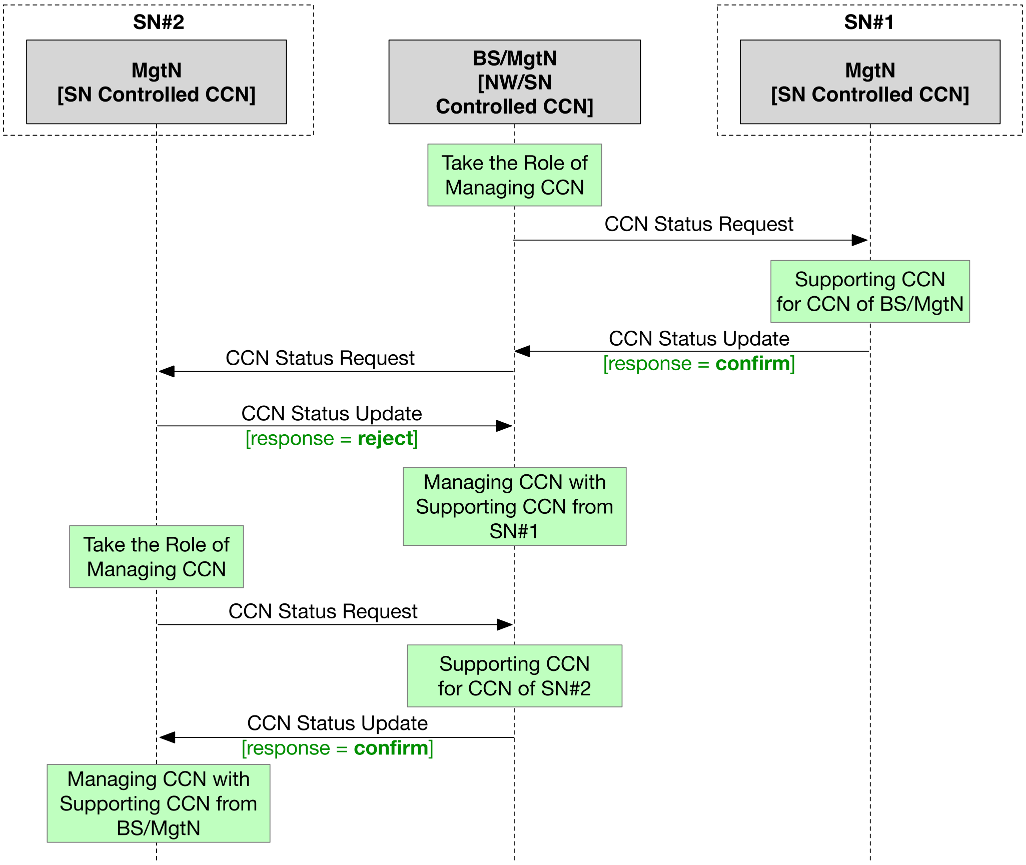


Figure 74 Managing and supporting CCN negotiations MSC using a request-based status update approach [23].

After the CCN negotiations have finished, the actual offloading can be requested by the ONs towards their CCN whether it being the Managing or a Supporting CCN. A Supporting CCN can operate in a non-transparent or a transparent mode. Meaning that in the non-transparent mode, the Supporting CCN evaluates the requested compute tasks and available compute resources within the local SN and decides which tasks can be locally offloaded and which shall be forwarded to the Managing CCN.

Whereas, in the transparent mode, the Supporting CCN simply passes all requested compute tasks and available compute resources within its SN to the Managing CCN, which in turn takes full control on the distribution of the compute offload tasks among all the available compute resources. Figure 75 shows a message sequence chart highlighting the messaging exchange between the different Supporting CCNs and the Managing CCN, where each CCN decides whether to operate in non-transparent or transparent mode.

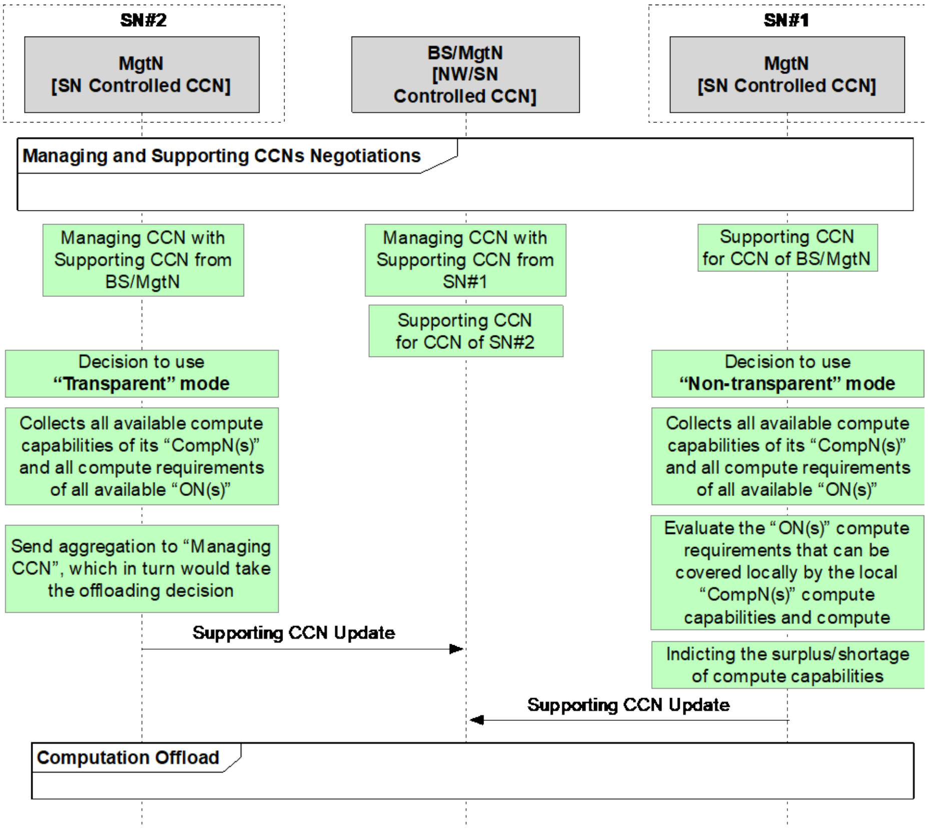

SN#2
MgtN
[SN Controlled CCN]
BS/MgtN
[NW/SN
Controlled CCN]
SN#1
MgtN
[SN Controlled CCN]
Managing and Supporting CCNs Negotiations
Managing CCN with Supporting CCN from BS/MgtN
Managing CCN with Supporting CCN from SN#1
Supporting CCN for CCN of BS/MgtN
Supporting CCN for CCN of SN#2
Decision to use "Transparent" mode
Decision to use "Non-transparent" mode
Collects all available compute capabilities of its "CompN(s)" and all compute requirements of all available "ON(s)"
Collects all available compute capabilities of its "CompN(s)" and all compute requirements of all available "ON(s)"
Send aggregation to "Managing CCN", which in turn would take the offloading decision
Evaluate the "ON(s)" compute requirements that can be covered locally by the local "CompN(s)" compute capabilities and compute
Indicting the surplus/shortage of compute capabilities
Supporting CCN Update
Supporting CCN Update
Computation Offload

Figure 75 Supporting CCN mode update procedure MSC [23].

4.1.4 Summary

The general use case of distributed compute has been addressed. The procedural framework has been presented for enabling both local and decentralized distributed compute offloading in SNs. For the local compute offload, the SN CCN Negotiations procedure within the SN as well as MgtN controlled and direct ON to CompN offload procedures have been introduced. This framework is complete since it covers the CCN selection along with the possible ways of enabling the offloaded, i.e. via the MgtN or directly between the ON and CompN. For the decentralized compute offload, transparent and non-transparent supporting CCN modes have been defined on top of the Managing and Supporting CCN role defined in D2.4 [3]. Finally, the negotiations procedure between the different CCNs across the different SNs and network entities to choose the managing CCN and supporting CCNs within any setup have been presented. With this extension to the procedural framework, the managing CCN is enabled to perform the ON and CompN pairing, even when these nodes belong to different SNs.

4.2 QUALITY OF COMPUTE SERVICE (QoCS) FRAMEWORK FOR SUBNETWORKS

4.2.1 Introduction

In 3GPP, the Quality of Service (QoS) framework [15] ensures end-to-end performance guarantees for specific applications and services, both for downlink and for uplink. The QoS framework is based on QoS flows, which present the finest granularity of QoS treatments in the 5G network. Each QoS flow has a unique QoS flow identifier (QFI), which identifies its QoS characteristics. Based on the specific 5G QoS identifier (5QI), which identifies the QoS characteristics of that QoS flow, RAN provides corresponding resources and prioritizes the scheduling transmissions to satisfy the required QoS of each QoS flow [15]. 5G QoS management framework is designed to address only communication requirements (i.e., throughput, delay, packet error rate, etc.) of different services [15], but not tailored to support computation services, which would guarantee the end-to-end communication and computation performance when UEs with limited computation, memory, storage, or power utilize the available external NW or UE resources.

The convergence of communication and computation is envisioned to be a driving force for realizing emerging computation-intensive and delay-sensitive applications in the next generation of mobile networks such as: AR/VR/XR, autonomous driving, smart manufacturing, online gaming, as well as different AI/ML operations supporting the mentioned applications including model/data sharing, split training and inference and distributed and federated learning. Moreover, to fully utilize the computation offloading capability in the SN architecture and to support computation requests with different resources and performance requirements, a Quality of Computation Service (QoCS) framework is needed. Accordingly, signalling procedures for managing the selection and signalling of the QoCS requirements and the reservation of necessary resources must be introduced. Moreover, the existing QoS and QoCS management solutions in mobile networks are mainly controlled by the Core Network (CN) and cannot be applied readily to the SN architecture since SNs need to perform in a device-controlled manner independently from the CN.

In the following sections a QoCS framework to support both communication and computation within a SN, between SNs, and between the SN and the overlay 6G network is introduced. It comprises of UE-centric, and network assisted frameworks for QoCS support in SNs. Furthermore, novel SN QoCS parameters and characteristics to fulfil required computation requirements are introduced. Finally, high-level procedures to support SN QoCS for local SN and decentralized compute offload are presented.

4.2.2 Background

It is assumed that sn-CCNs of different SNs perform a negotiation procedure among themselves and select a *managing CCN (mgt-CCN)* and the rest of the CCNs act as *supporting CCNs (sup-CCNs)* as described in Section 4.1. There may exist a NW infrastructure that provides communication and/or computation support for the SNs. In the case that the NW infrastructure is available and provides computation resources as well as communication, the network CCN is also involved in the negotiation procedure and most probably the network CCN is selected as a mgt-CCN. If the NW provides only communication between SNs, the NW CCN is not considered in the negotiation procedure.

The sn-CCN in each SN is responsible for mapping the compute request received from an ON to CompN(s) given the session management subscriptions and received CompN capabilities information. If there is no CompN within the SN that can support the requested compute workload, the sn-CCN routes the request to the mgt-CCN (non-transparent mode). Otherwise, in a transparent mode, the sn-CCN forwards all compute requests and capacity to managing CCN, which would then take the decision.

4.2.3 Subnetwork QoCS characteristics

A QoS flow contains data where the QoS is described by a parameter 5QI. These 5Q QoS characteristics include *Resource Type (Guaranteed Bit Rate (GBR), Non-GBR, Delay critical GBR)*, *Priority level, Packet Delay Budget (PDB), Packet Error Rate (PER), Averaging Window and Maximum Data Burst Volume (MDBV) [15]*. The 5G QoS characteristics should be understood as guidelines for setting (UE or NW) node specific parameters for each QoS flow, e.g., for RAN protocol configurations. Standardized or pre-configured 5G QoS characteristics are indicated through the single 5QI value.

In the proposed QoCS framework, the set of communication characteristics is extended and the following computation characteristics to control the QoCS-related operations between network nodes are proposed:

- *Computation Resource Type*, which determines if dedicated compute resources are permanently allocated to a flow by an admission control algorithm and if they can be *Guaranteed Computation Delay (GCD)* and *Non-GCD*.
- *Default Computation Priority Level*, which indicates the priority for scheduling compute resources.
- *Interaction type* presenting the number of iterations required by the task, e.g., multiple iterations for FL/Distributed Learning-based applications, or single shot for photo enhancement at smartphone. *Interaction type* can be *Single shot* (workload request and info sent by ON, and result received from CompN) or *Iterative* (multiple iterations of workload info submission and result reception until the final result is obtained).
- *Number of UEs* [optional], presenting the number of UEs cooperating on the computation task, e.g., for a FL service, the number of UEs participating in the local training.
- *Reliability,* which indicates the required reliability level for the chosen CompN for the requested computation task and can be simply modeled as a confidence value (real number between 0 and 1), or a qualitative value of high/low reliability.
- *Privacy/security*, which indicates the required privacy/security level for the chosen CompN for the requested computation task. Similarly to the *Reliability*, it can be modeled simply as a trust value (real value between 0 and 1), or a qualitative value.
- *Numerical precision*, which is a real value determining the computation numerical precision requirement.
- *Delay violation probability/rate* [optional], presenting a real value in between 0 and 1, which determines the tolerable computation (or computation and communication) delay violation. It can be interpreted as the percentage of the computation results experiencing a delay exceeding the flow computation delay budget.
- *Default averaging window*, presenting the window over which the computation delay is averaged.

A combination of the computation characteristics is mapped to a novel identifier, namely *Subnetwork QoCS Identifier* (*SN-QCI*). The QoCS flow types with the corresponding QoS and QoCS parameters are given in Figure 76.

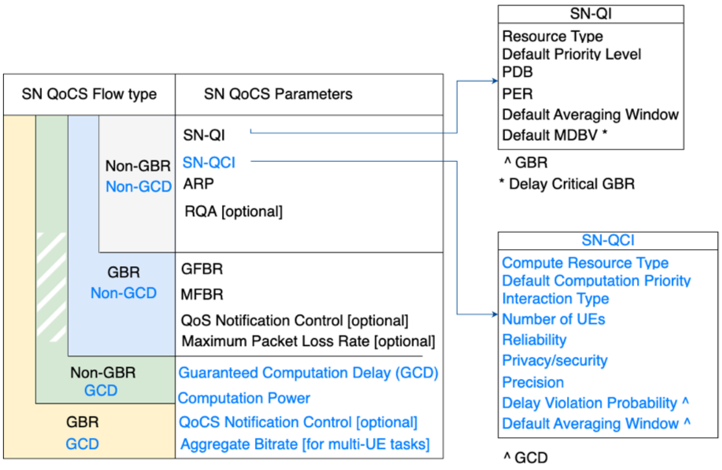


Figure 76 QoCS flow types: separate SN-QI and SN-QCI.

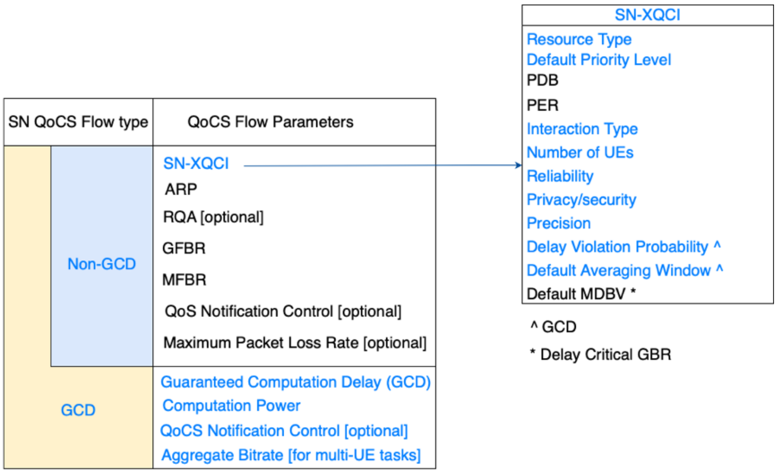


Figure 77 QoCS flow types: single SN-XQCI.

Here, besides the 5G QoS parameters, such as *Allocation and Retention Priority (ARP), Reflective QoS attribute (RQA), Guaranteed Flow Bit Rate (GFBR), DL and UL Maximum Flow Bit Rate (MFBR) for UL and, SN QoS Notification Control (QNC), DL and UL Maximum Packet Loss Rate for UL and DL* [15], the novel QoCS parameters are introduced:

- *SN QoS Indicator (SN-QI),* a scalar used to refer to the communications characteristics.
- *SN QoCS Indicator (SN-QCI),* a scalar used to refer to the QoCS characteristics described above.
- *Guaranteed Computation Delay (GCD),* which presents the guaranteed computation delay to be provided by the network.
- *Computation Power*, indicating the estimated power required for the computation task.
- *SN Compute Notification Control (CNC)* [optional], which indicates whether notifications are requested from the MgtN when the "GCD can no longer (or can again) be guaranteed" for a given GCD QoCS.
- *Aggregate Bitrate* [multi-UE tasks, optional], which is the aggregate flow bitrate required for multi-UE SN QoCS flows.

If a service does not require computation, a default SN QoCS (i.e., SN-QCI = 0) is used and only SN-QI is configured. Otherwise, both SN-QI and SN-QCI are configured to meet the communication and computation service requirements, respectively.

Another option would be for the SN QoS and SN QoCS characteristics to be mapped to a single indicator *SN-XQCI*, shown in Figure 77, while the flow type can be configured either for QoS or QoCS service, by signalling flow type (FT) parameter during the SN flow establishment/modification.

4.2.4 Procedures for QoCS signalling for Local Compute Offload

As introduced in 4.1.2, in the case of a local compute offload, the computation offload is kept within the local SN, such that the ON offloads its tasks only to the available CompN(s) in the MgtN and/or HC device(s) within the local SN without NW involvement. The local MgtN’s sn-CCN is assumed to be the default compute offload controlling entity, without any involvement from any other CCN entities of the NW and other SNs. The sn-CCN creates SN QoCS Rules (SN-CQFI, ON ID, CompN ID) and Profiles (SN-QCI, QoCS parameters) and SN QoCS flow(s) are established between ON, CompN and sn-CCN. Moreover, the sn-QoCS flow can be established also directly between ON and CompN(s).

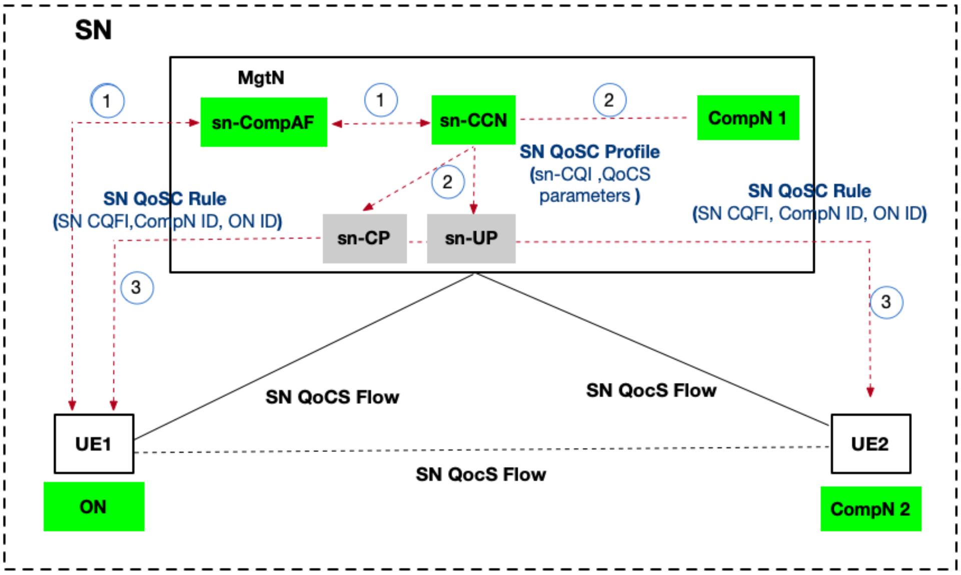


Figure 78 Local SN compute offload.

High-level signalling of SN QoCS is illustrated in Figure 78. The *Offloading UE* or *SN Application* may request for QoCS directly from the sn-CCN by sending the computing service requirements (UE ID, bandwidth requirement, SDF description, compute task request), e.g., via application layer signaling towards an *sn-ComputeAF* (an AF of the required SN Application) (1). The sn-CCN, based on stored information about CompN(s) capabilities (ID, capacity, location, mobility status, addresses, resource status), translates this compute request and allocates a number of CompNs for a given task (2). The sn-CCN then generates QoCS parameters (compute/communication requirements) and sends SN QoCS profile(s) towards the sn-CP, which then communicates QoCS rules to CompN and ON (3).

4.2.5 Procedures for QoCS signalling for Decentralized Compute Offload

As presented in 4.1.3, in the case of decentralized compute offload, the computation offload is distributed among different NW and/or SN nodes. The ON offloads its task to any available CompN(s) (i.e., neighbouring SN(s), BS(s), CN, cloud server(s), etc.) with or without NW involvement. One of the involved sn-CCN will act as the mgt-CCN and all other CCNs involved (i.e., belonging to the other SNs) will act as sup-CCN(s). Two variants of decentralized compute offload are considered, with and without utilizing the overlay NW communication resources.

In the first variant, shown in Figure 79, the SN registers to the NW to use the NW’s communication resources. The mgt-CCN creates SN QoCS rules and Profiles and SN QoCS flow(s) are established between ON and sn-CCNs, as well as between CompN and sn-CCN. Moreover, the mgt-CCN sends request for NW’s communication resources, and based on the request and the available communication resources, the NW establishes communication QoS flow between the mgt-CCN, the NW and the sup-CCN.

High-level signalling of SN QoCS is illustrated in Figure 79 as well. The *Offloading UE* or *SN Application* may request for QoCS directly from the sup-CCN, which, in turn, forwards this request towards the mgt-CCN (1). The mgt-CCN, based on stored information about CompN(s), translates this compute request into a compute capacity demand and in turn and allocates a number of CompNs for a given task. The mgt-CCN then generates QoCS parameters (compute/communication requirements) and sends SN QoCS profile(s) towards the sn-CP (2), which then communicates QoCS rules to CompN and ON for establishing the QoCS flow (3). The mgt-CCN sends the request for communication resources to the NW (4), and the NW sends QoS rules to the mgt-CCN and the sup-CCN for establishing QoS flows between sup-CCN and NW and NW and mgt-CCN (5).

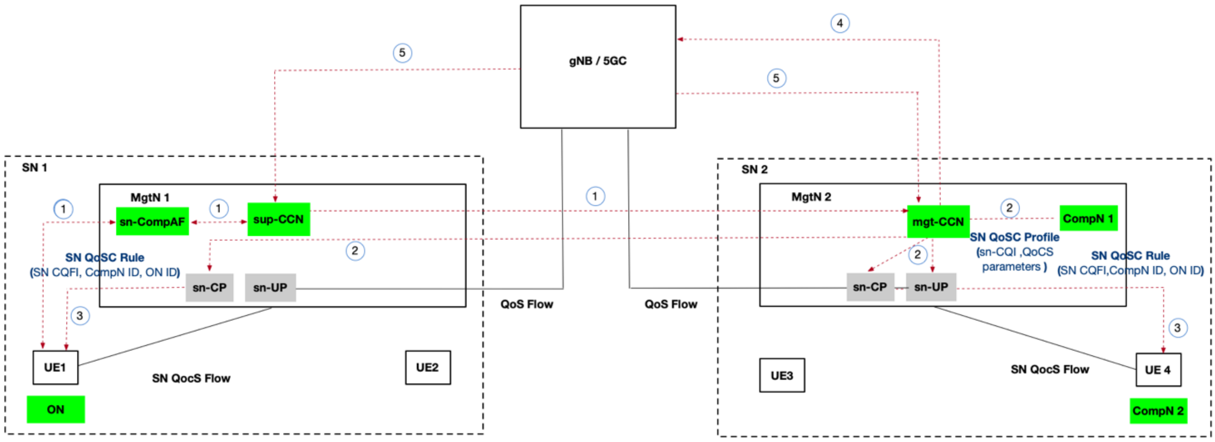


Figure 79 Decentralized Compute Offload Option 1: Involvement of NW communication resources.

In the second variant, there is no NW involvement in offering either computation or communication resources for the compute offload procedure, therefore involving direct sup-CCN and mgt-CCN communication. Here, the mgt-CCN creates SN QoCS rules and profiles and SN QoCS flow(s) are established between ON, CompN and sn-CCNs as shown in Figure 80.

The *Offloading UE* or *SN Application* may request for QoCS directly from the sup-CCN, which, in turn, forwards this request towards the mgt-CCN (1). The mgt-CCN, based on stored information about CompN(s) translates this compute request into a compute requirement and in turn allocates several CompNs for a given computation task. The mgt-CCN then generates QoCS parameters (compute/communication requirements) and sends SN QoCS profile(s) towards the sn-CP (2). QoCS rules for QoCS flow are communicated from sn-CP in MgtN 1 to ON and from sn-CP in MgtN 2 to CompN 2 (3), while the SN QoCS flows are established via sn-UP.

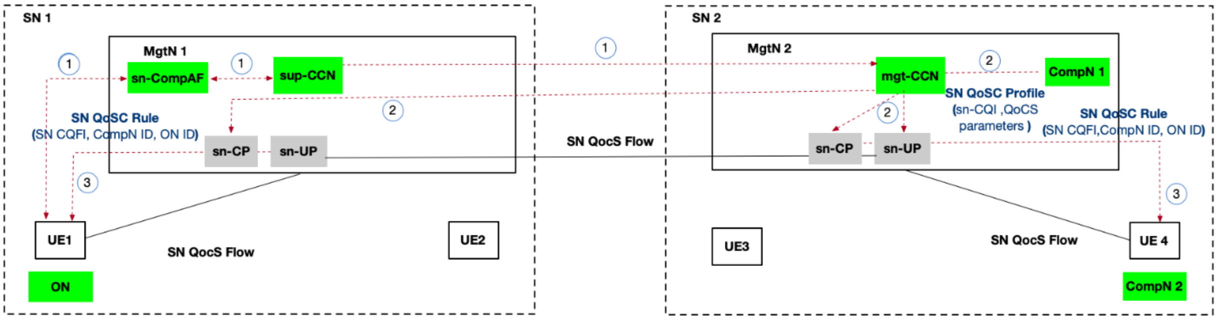


Figure 80 Decentralized Compute Offload Option 1: Without the involvement the NW communication resources.

4.2.6 Summary

In this section, a UE-centric QoCS framework has been introduced to fully enable computation offloading capability in SN architecture and to support computation requests with different resource and performance requirements. To control the QoCS-related operations between network nodes, a set of new SN-specific QoCS characteristics, such as computation resource type, interaction type, compute precision, and privacy has been introduced. Moreover, to determine SN QoCS Flow types, novel SN-specific QoCS parameters, including SN QoCS Indicator and computation power among others, have been proposed. Finally, complete signalling procedures have been presented in order to support QoCS in SNs for both local and decentralised compute offloading cases.

4.3 JOINT TASK AND COMMUNICATION SCHEDULING FOR DEPENDABLE SERVICE LEVEL PROVISIONING

4.3.1 Introduction

Beyond 5G (B5G) and 6G networks are envisioned as a 'network of networks' (NoN) ecosystem, integrating diverse communication networks to enable seamless and ubiquitous connectivity [37]. This includes subnetworks deployed at the deep edge of the network for local communications (see [37], [38] and 3GPP TSG-SA WG1, e.g., S1-240121, S1-244238). Subnetworks are composed of different types of IoT devices (e.g. sensors and actuators) that can seamlessly interconnect either locally – via sidelink or direct communications or with a wide-area cellular network for providing cost-effective service delivery for applications with diverse requirements while supporting distributed processing for autonomous local data management. The B5G and 6G vision goes beyond pure communication systems, aiming to sustainably integrate computing, communication and intelligence into a unified system capable of supporting the ever-growing demands across an IoT-edge-cloud continuum [39][40]. This continuum provides a programmable computing infrastructure across IoT devices, edge and cloud nodes that expands capabilities and flexibility for dynamically deploying applications and network services while adapting deployments to variable demands.

Realizing the potential of the IoT-edge-cloud continuum requires efficient task offloading and resource allocation strategies to dynamically distribute tasks across resources in the continuum. These strategies should consider the requirements from various types of applications including emerging ones like critical vertical applications in fields such as industrial automation, cyber-physical systems, healthcare, or autonomous mobility. Many of these applications demand high dependability and deterministic service levels, which existing networks cannot easily provide. However, they can benefit from the distributed computing resources within the IoT-edge-cloud continuum. In fact, deterministic communications and networking have been identified as key enablers in 6G to support emerging and critical vertical applications at scale [41]. Future networks and systems must efficiently scale while maintaining the required dependability and deterministic service levels, even as the number of connected devices, computational demands, spectrum constraints and stringent communication requirements continue to grow. Achieving this scalability in a 6G-based NoN ecosystem operating an IoT-edge-cloud continuum depends on the effectiveness of task offloading and resource allocation strategies in efficiently managing communication and computing resources in the continuum.

This study advances the state of the art by demonstrating that a deterministic approach to task offloading and resource allocation not only ensures the required deterministic service levels but also scales more effectively than existing task offloading and resource allocation strategies. To this end, we compare the scalability of a deterministic policy with a state-of-the-art policy that seeks minimizing task execution time. Our results show that, by flexibly managing task completion deadlines, a deterministic approach to task offloading and resource allocation achieves a more balanced workload and resource distribution across the continuum. This, in turn, improves the system's ability to meet task execution deadlines for a larger number of tasks and nodes, enhancing overall scalability. Our evaluation further demonstrates that a deterministic policy leads to higher task completion rates, improved fairness across the system, and greater adaptability to variations in computing and communication resource utilization and conditions.

4.3.2 State of the Art

The integration of computing and intelligence enables a Local-edge-cloud continuum, which provides opportunities to balance task allocation between local nodes in the subnetworks and remote servers at the edge or in the cloud. While offloading tasks to remote servers can reduce computing or processing latency, it introduces communication latency due to data transmission from local nodes to remote servers. On the other hand, relying on local nodes in the subnetworks may increase computing latency because of their generally lower processing power compared to edge nodes or cloud servers.

The opportunities provided by the Local-edge-cloud continuum have spurred significant research in recent years to design task offloading and resource allocation mechanisms that optimize latency in distributed computing environments. Most existing contributions focus on minimizing total latency, which includes both computing and communication latency, by finding an optimal balance between local computation and task offloading to remote servers. Studies have explored this balance, highlighting that decisions regarding how much data should be processed locally versus offloaded remotely depend on factors such as communication bandwidth and available processing power.

However, as highlighted in D4.2 [4], deterministic service level provisioning requires a different approach. The preliminary framework introduced in D4.2 emphasized the need to jointly allocate computational and communication resources while considering the bounded deadlines of various tasks. This deliverable extends that work by further refining the optimization models and evaluating their performance under different workload conditions.

4.3.3 Architecture and System Model

Subnetworks can be integrated in the Local-edge-cloud continuum for local connectivity [37]. Given the critical nature of some of these local connectivity scenarios, subnetworks must be able to operate standalone or connected to a parent cellular network, which can support the operation and configuration of subnetworks. Figure 81 depicts the envisioned Local-edge-cloud network architecture integrating subnetworks alongside communication and computing domains (edge and cloud nodes). The subnetwork consists of Subnetwork Elements (SNEs), Low Capability (LC) units, and High Capability (HC) units. The HC serves as the central hub within the subnetwork and as a gateway between the subnetwork and the parent network. The HC unit can handle most computationally intensive tasks and offer computing resources to other units within the subnetwork. The LC unit has reduced networking and computing capabilities compared to the HC unit. It can act as an aggregator or gateway between SNEs and the HC but may not have direct access to the parent network. SNEs are computationally constrained.

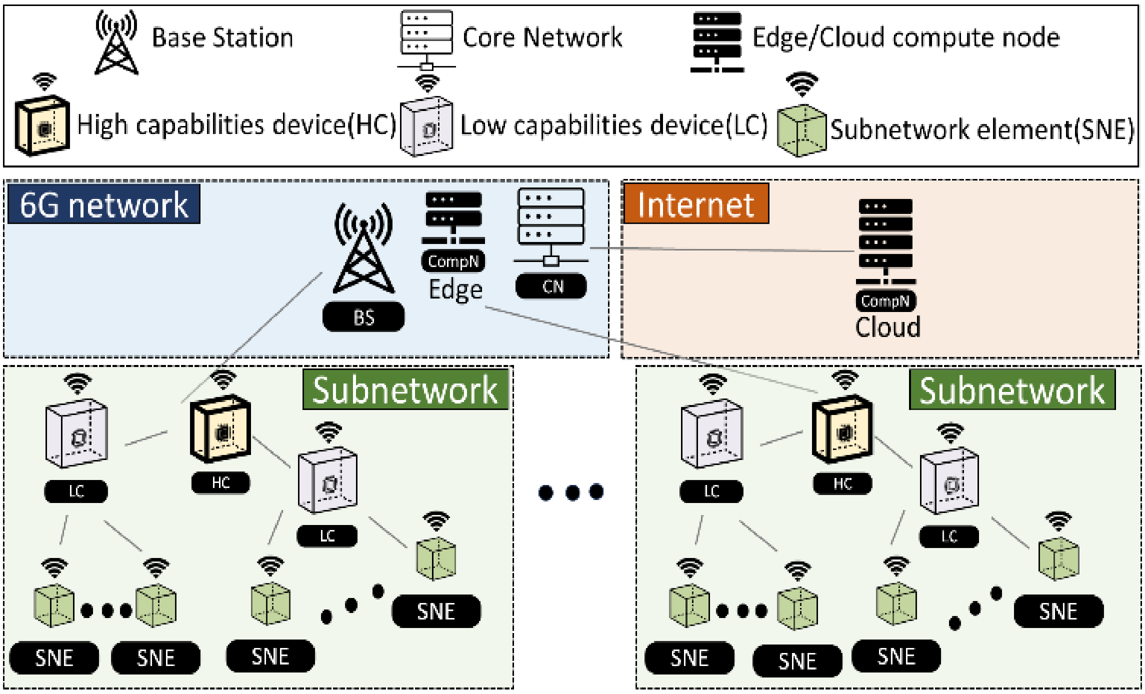


Figure 81 IoT-edge-cloud architecture.

Figure 82 illustrates an example of in-vehicle subnetworks using a zonal E/E architecture as envisioned in the transition towards software defined vehicles and autonomous driving. The in-vehicle subnetwork interconnects all devices and automotive domains with the HPCU (High-Performance Computing Unit) acting as the HC unit, 4 Zone ECUs (ZoneElectronic Control Units) functioning as LC units, and sensors and actuators serving as SNEs.

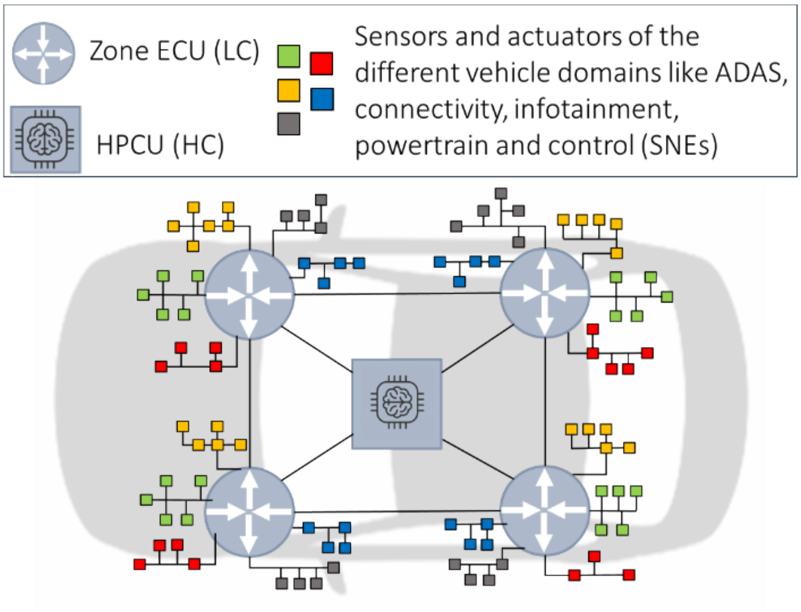

Zone ECU (LC)
HPCU (HC)
Sensors and actuators of the different vehicle domains like ADAS, connectivity, infotainment, powertrain and control (SNEs)

Figure 82 In-vehicle subnetworks.

The 6G NoN vision integrates multiple networks and domains in a unified framework. This includes subnetworks for local IoT communications, e.g. at factories for communications between robots or collaborative robots (cobots), as well as within and between autonomous vehicles [37], [38]. This integration provides the possibility to establish an IoT-edge-cloud continuum where tasks can be seamlessly offloaded across the continuum. Figure 81 shows the IoT-edge-cloud architecture that integrates subnetworks for local connectivity at the deep edge. Each subnetwork comprises three main components: Subnetwork Elements (SNEs), Low Capability (LC) units, and a High Capability (HC) unit [38].

In this IoT-edge-cloud architecture, we consider a collection of tasks $f_{i,n}$ ($i \epsilon\{1, \dots , I\}$) generated within the subnetwork n ($n \epsilon\{1, \dots , N\}$) at time instant $t_{i,n}$. Tasks can be generated by SNE_n, LC_n, and HC_n of the subnetwork n. Each task $f_{i,n}$ is defined by its computing demand $c_{i,n}$ and associated size $s_{i,n}$. When a task is offloaded to a processing unit different from where it was generated, the processed result, which has a reduced size $s'_{i,n}$ compared to the original size $s_{i,n}$, must be transmitted back to its source unit. Each task has a deadline $T_{i,n}^{max}$, which indicates the maximum time available to complete processing. Computing units in different subnetworks have different processing capacities, denoted by P_x, where x refers to the type of processing unit. $x_s \epsilon\{LC_n, HC_n\}$ represents local processors of the subnetworks, while $x_p \epsilon\{Ed, Cl\}$ refers to the edge (Ed) and cloud (Cl) units. The time required to process a task $f_{i,n}$ on a computing unit $x \epsilon\{x_s, x_p\}$ is given by the following equation:

$$t_p^{i,n} = \frac{c_{i,n}}{P_x}.$$

We assume an Orthogonal Frequency Division Multiple Access (OFDMA) radio access interface for the wireless links within subnetworks and for connecting subnetworks to the parent network. Subnetworks have a dedicated communication band with a bandwidth of BW_s that does not overlap with the band used for communication with the wide-area cellular network, which has a bandwidth BW_p. Inter-subnetwork interference is not considered based on recent subnetwork channel characterization measurements [42] that demonstrate the possibility to isolate subnetworks through well-planned and characterized environments (e.g. introducing directive communications and reconfigurable intelligent surfaces (RIS)), and the high penetration losses of material within the subnetworks (e.g. within vehicles). The subnetworks' bandwidth BW_s is divided into K_s orthogonal communication resources, which can be reused within different subnetworks. $BW_p$is divided into K_p orthogonal resources, and these resources are shared among the N subnetworks for their connectivity with the wide-area cellular network. In accordance with the 3GPP 5G NR standard, our model adopts a subcarrier spacing (SCS) of 30 KHz [43] and a time slot duration of 0.5 ms [37]. The data rate available at any given time for communication resource $k \epsilon\{K_s, K_p\}$ in link $l \, \epsilon\{1,2,..,L\}$, either within the subnetwork n or from the subnetwork n to the wide-area network, is denoted as $r_{l,n}^{(k)}(t)$ and can be expressed as [44] :

$$r_{l,n}^{(k)}(t) = BW_k \cdot log_2\left(1 + \gamma_{l,n}(t)\right)(1 - BER),$$

where BW_k represents the bandwidth of the communication resource k,$\gamma_{l,n}(t)$ denotes the Signal-to-Interference plus Noise Ratio (SINR) at time t of the link l, and BER is the bit error rate, which depends on the modulation and coding scheme employed in the communication resource k. Similar to system model with one subnetwork, the total data rate of a communication link l is calculated as the sum of the data rates for all communication resources k utilized in the link:

$$r_{l,n}(t) = \sum\nolimits_k r_{l,n}^{(k)}(t).$$

When a task $f_{i,n}$ requires offloading, the transmission time over different communication links $l_i \in \{1,2,..,L\}$ is determined as:

$$t_c^{i,n} = \sum\nolimits_{l_i} \frac{s_{i,n}}{r_{li,n}(t)}.$$

Similarly, the transmission time over communication links $l_i \in \{1,2,..,L\}$ for the processed result of a task $f_{i,n}$ with size of $s'_{i,n}$ can be expressed as $t'^{i,n}_c$ and is computed following $t_c^{i,n}$ using $s'_{i,n}$ instead of

$s_{i,n}$. The total time $T_{i,n}$ required to execute a task $f_{i,n}$ generated within the subnetwork n, includes the communication time for moving the task from its source to the processing unit ($t_c^{i,n}$), the processing time at the computing unit ($t_p^{i,n}$), and the communication time for returning the processed result ($t'^{i,n}_c$):

$$T_{i,n} = t_c^{i,n} + t_p^{i,n} + t'^{i,n}_c \,.$$

4.3.4 Deterministic Resource Allocation Algorithm

Existing task offloading and resource allocation schemes mostly focus on minimizing the total execution time of tasks. However, this approach can put excessive strain on the network, generating peaks in computing and communication demands that overload certain parts of the network. Such overloads can create bottlenecks that may unnecessarily delay the timely execution of certain tasks. In contrast, we advocate for a deterministic approach, where communication and computing resources are jointly allocated and managed to meet tasks' specific bounded latency deadlines instead of simply minimizing execution time. This approach leverages diverse tasks' deadlines to increase the number of satisfactorily executed tasks (i.e., their execution time T_i is lower than their deadlines T_i^{max}) without overburdening the network's computing and communication resources. The proposal is designed and evaluated within the IoT-edge-cloud continuum framework described in Section 4.3.3, where tasks can be processed locally or offloaded to edge or cloud servers. In this study, local processing refers to processing within a subnetwork, and tasks may be offloaded within subnetwork units, as previously described.

The objective function for our deterministic joint task offloading and resource allocation proposal is defined as:

$$min \sum\nolimits_i K\left(\frac{T_i}{T_i^{\max}}\right),$$

where T_i is the execution time of task i, T_i^{max} is the deadline of task i, and K is a penalty function defined as:

$$K(x) = \begin{cases} 0, & 0 \leq x \leq 1, \\ M, & x \geq 1, \end{cases}$$

where x = 0 represents the task's generation time t_i, x = 1 represents its deadline T_i^{max}, and M is a high positive constant value. The objective is to minimize the number of tasks where the execution time T_i exceeds the deadline T_i^{max}. The penalty function assigns a high penalty to tasks that exceed their deadlines, while tasks completed before their deadlines incur no penalty regardless of their specific execution time. The objective function aims to ensure task execution within bounded deadlines (i.e., deterministic) without placing unnecessary strain on network resources.

The objective function includes four additional constraints. First, the allocation of tasks to processing units is binary, meaning each task is assigned to a single computing unit and cannot be split across multiple units. This is expressed as:

$$\sum\nolimits_{j=1}^{J} a_i^{(j)} + \sum\nolimits_{q=1}^{Q} a_i^{(q)} + a_i^{(c)} = 1, \forall i,$$

where J is the number of local computing units in the subnetwork (i.e., the sum of LC and HC units) and Q is the number of edge computing units. $a_i^{(j)}$, $a_i^{(q)}$ and $a_i^{(c)}$ are binary variables equal to 1 if task i is allocated to the local computing unit j, the edge unit q, or the cloud unit c, respectively, and 0 otherwise. In line with OFDMA, the second constraint establishes that communication resources can be used by a single communication link at a time, ensuring there is no interference within the subnetwork or in the parent network. This is expressed as:

$$\sum_{i=1}^{I} b_{l,i}^{(k)} = 1, \ \ \forall k, l,$$

where I is the number of tasks, and $b_{l,i}^{(k)}$ is a binary variable equal to 1 when communication resource k is allocated to transmit task i in link l. The third constraint is that the transmission rate for all tasks utilizing one link must not exceed the maximum possible data rate of that link:

$$\sum_{i=1}^{I} r_{l,i}(t) \leq r_l(t), \, \forall l,$$

where $r_{l,i}(t)$ is the data rate of link l for transmitting task i and $r_l(t)$ is the maximum possible data rate of link l. The fourth constraint is that the total processing workload of tasks assigned to a computing unit over a given time interval must not exceed the maximum processing capacity of that unit:

$$\sum_{i=1}^{I} c_i a_i^{(x)} \leq C_x^{max}, \forall x.$$

The optimization problem is NP-complete, and its computational complexity increases exponentially as the number of computation and communication resources grows. We have implemented a genetic algorithm in MATLAB to resolve the resource allocation problem that uses 10 generations/iterations. The implemented genetic algorithm considers a population size of 1000 for each generation and introduces mutation to converge to an (near) optimal solution [45].

4.3.5 Priority-based task offloading

In deterministic approach to task offloading and resource allocation, communication and computing resources are jointly managed to ensure that tasks are executed within their specific bounded latency deadlines.

The objective function for our deterministic joint task offloading and resource allocation proposal considering priority is defined as:

$$min \sum_{i} \Gamma\left(\frac{T_i}{T_i^{\max}}\right)$$

where T_i is the execution time of task i, T_i^{max} is the deadline of task i, and $\Gamma(.)$ is a penalty function defined as:

$$\Gamma(x) = \begin{cases} 0, & 0 \leq x \leq 1, \\ P_i M, & x \geq 1, \end{cases}$$

where x = 0 represents the task's generation time t_i, x = 1 represents its deadline T_i^{max}, P_i is its priority factor, and M is a high positive constant value. The penalty function increases the penalty for high-priority tasks by incorporating a priority factor P_i , where higher values of P_i (for high priority tasks) result in greater penalties for exceeding deadlines. This ensures that critical tasks are given higher importance in resource allocation and offloading decisions, reinforcing the deterministic approach's goal of meeting strict timing constraints.

4.3.6 Performance Evaluation of Deterministic Resource Allocation

4.3.6.1 Evaluation Scenario

The IoT-edge-cloud continuum scenario considered for evaluation includes a subnetwork with 5 local processors (1 HC processor and 4 LC processors) and 35 SNEs, 1 edge node and 1 cloud server. Based on features of commercial off-the-shelf products [46], the processing power of the units are: 2.5 GHz (2.5G operations per second) for LC, 5 GHz for HC, 70 GHz for the edge node, and 150 GHz for the cloud server. This study does not focus on a specific IoT application. Instead, we consider that $\alpha_{SNE} = 60\%$, $\alpha_{LC} = 20\%$, and $\alpha_{HC} = 20\%$ of the tasks are generated by the SNE, LC, and HC, respectively. The tasks are generated randomly, following a uniform distribution, throughout the simulation time. Tasks can be allocated to any unit across the continuum. The processing workload and size of the tasks are modeled as uniform random variables [47] within the ranges (20, 50) M cycles [48] for the workload and (0.75, 2.25) M bits for the size [49]. The size of the processed results for a task is set to 15% of the task size. Tasks are randomly assigned a deadline T_i^{max} following a uniform distribution between 20 ms and 100 ms. The penalty value M in the penalty function assigned to tasks that are not completed by their deadlines is 100. The bandwidths for intra-subnetwork links and for connecting to the parent network are 100 MHz and 50 MHz, respectively [43]. The intra-subnetwork links operate with an SINR of 30 dB, while the SINR for the connection between the subnetwork and the parent network varies between 3 dB and 27 dB.

We compare the performance of our proposed deterministic scheme against a random allocation and a state-of-the-art benchmark scheme. However, most of the comparisons focus on the deterministic and benchmark schemes as they outperform the random allocation. The random allocation scheme selects the computing unit for each task randomly and, if applicable, also randomly selects the communication resources of the link to reach that computing unit from those available until the task's deadline. The benchmark scheme ([47]) allocates tasks to computing units with the objective of minimizing the execution time of individual tasks based on the following objective function:

$$min \sum_i T_i,$$

where $\boldsymbol{T_i}$ is the execution time of task $\boldsymbol{i}$ as defined in section 4.3.3.

4.3.6.2 Results

Figure 83and Figure 84 compare the average ratio of satisfied tasks as a function of the number of tasks being executed. This ratio is defined as the proportion of tasks successfully completed before their deadlines relative to the total number of tasks. Results are shown for good SINR conditions in the connection to the parent network (27 dB) and bad conditions (3 dB), respectively. Figure 83 and Figure 84 demonstrate that our deterministic proposal achieves the highest ratio of satisfied tasks, regardless of the number of tasks or the channel quality conditions. As expected, the ratio of satisfied tasks decreases for both schemes as the number of tasks increases, due to limitations in available computing and communication resources. However, results clearly show that prioritizing task completion before deadlines (i.e., 'Deterministic'), rather than trying to minimize the execution time of individual tasks (i.e., 'Minimum'), enables the system to satisfactorily handle more tasks. The same trend is observed in Figure 85 and Figure 86, which plot the average ratio of satisfied tasks as a function of the average SINR for the connection between the subnetwork and the parent network. The average SINR of the intra-subnetwork wireless links is maintained at 30 dB. Results are reported for scenarios with 35 and 45 tasks, respectively, as these workloads approach the capacity limit of the system model, based on Figure 87. Figure 87shows the number of satisfied tasks as a function of the SINR when the total number of tasks

executed is 75. The figure reveals that the number of satisfied tasks does not exceed 52, even under good channel conditions. Figure 85 and Figure 86 show that the ratio of satisfied tasks increases as the link quality improves. This is because the data rate of the link increases with better SINR conditions due to the use of higher-order modulation and coding schemes. As the data rate increases, more tasks can be offloaded to the edge or cloud and be served within their latency limits. Figure 83 to Figure 86 show that the gains of the deterministic proposal over the benchmark scheme are more pronounced when communication or computing resources are more constrained, for instance, when the SINR to the parent network degrades or when the system approaches its capacity limit (i.e. between 45 and 55 tasks).

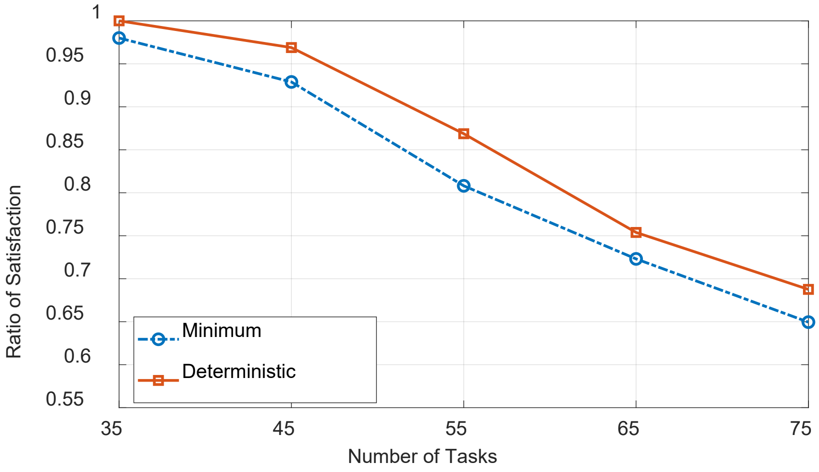


Figure 83 Average ratio of satisfied tasks as a function of the number of tasks for SINR= 27dB.

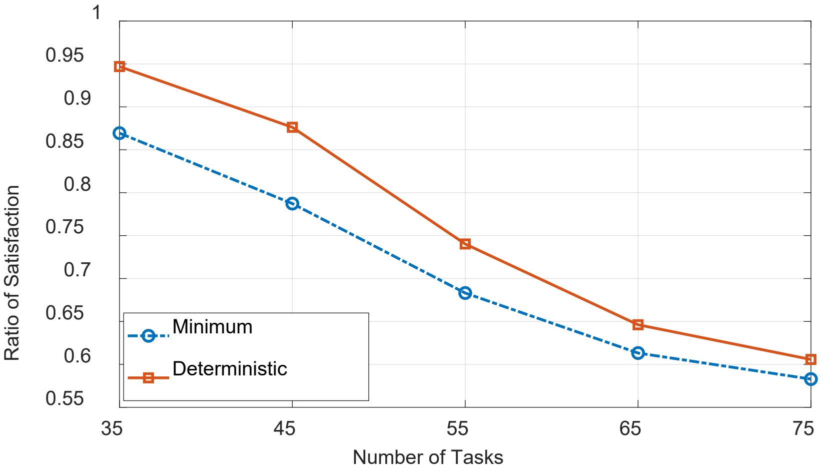


Figure 84 Average ratio of satisfied tasks as a function of the number of tasks for SINR= 0 dB.

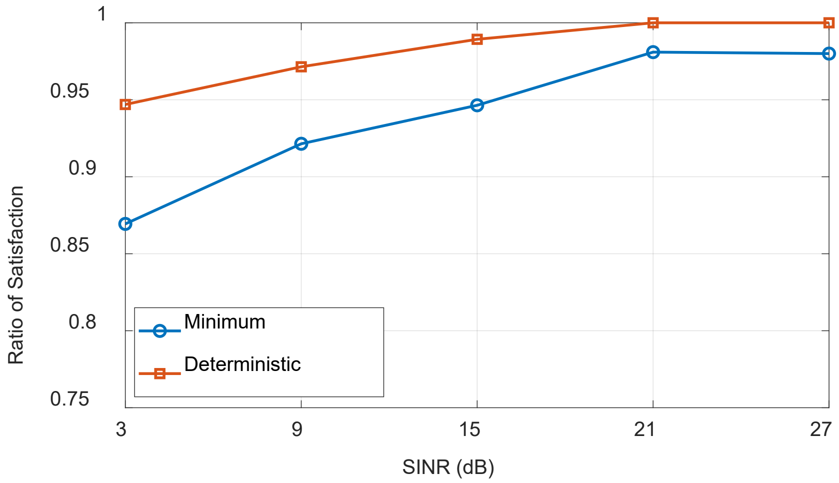


Figure 85 Average ratio of satisfied tasks as a function of the SINR when 35 tasks are executed.

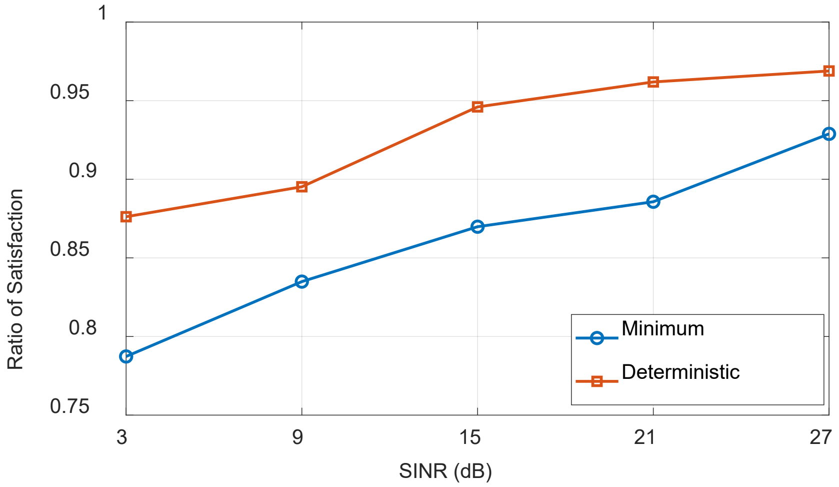


Figure 86 Average ratio of satisfied tasks as a function of the SINR when 45 tasks are executed.

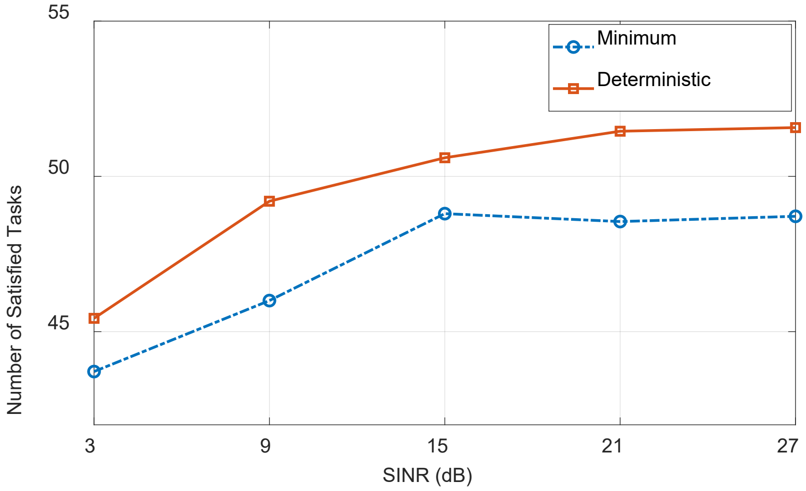


Figure 87 Number of satisfied tasks as a function of the SINR. The total number of executed tasks is 75.

Figure 88 depicts the normalized time budget, defined as the time remaining since a task is completed to its deadline, relative to the deadline. A normalized time budget of 0 indicates that a task has been successfully completed just at its deadline, while higher values indicate that the task was completed earlier than the deadline. This metric assesses which scheme prioritizes early task completion, and which one achieves more balanced completion times. The figure presents the results as a heat bar, showing the normalized time budget for different percentiles of tasks. Figure 88 reveals that the benchmark scheme, which aims to minimize the execution time of tasks, increases the number of tasks completed earlier compared to the deterministic scheme. For example, with the benchmark scheme, 80% of tasks are successfully completed before the normalized time budget reaches 0.6. In contrast, the deterministic proposal completes approximately 60% of tasks before the normalized time budget reaches 0.6. However, by flexibly exploiting varied task deadlines, the deterministic proposal can complete all tasks before their respective deadlines. In comparison, the benchmark scheme, focused on minimizing individual task execution times, can only successfully complete less than 90% of tasks under the conditions reported in Figure 88. Furthermore, the results show that the benchmark scheme and the random allocation require 12% and 72% more time, respectively, to complete all tasks compared to the deterministic proposal.

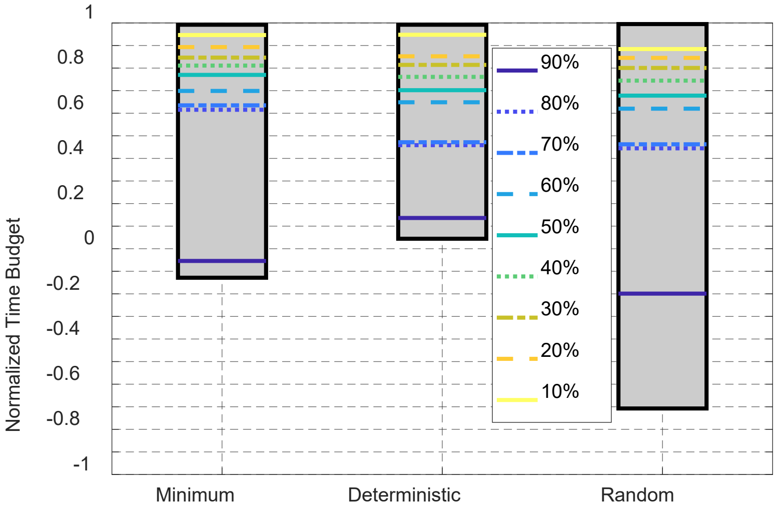


Figure 88 Normalized time budget for 35 tasks and SINR =27 dB.

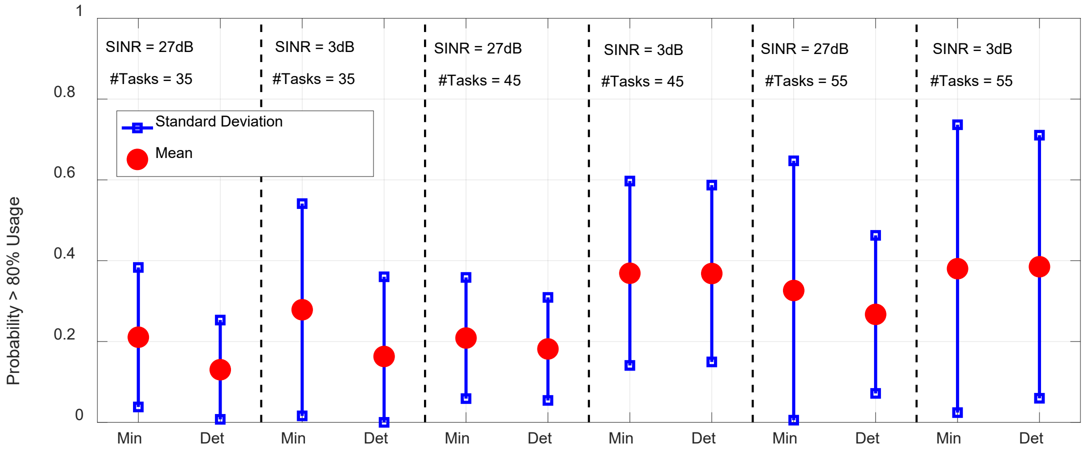


Figure 89 Average and standard deviation of the probability of saturating the use of communication resources for various SINR and tasks.

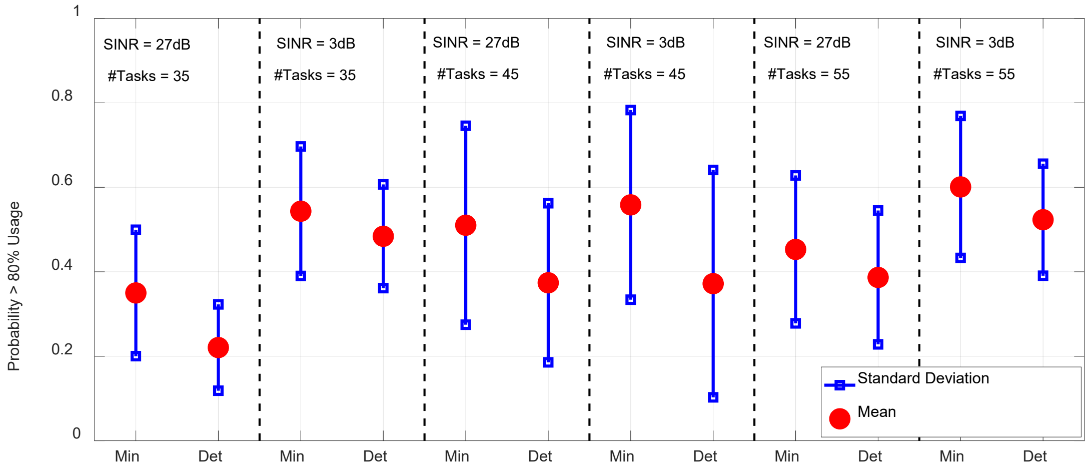


Figure 90 Average and standard deviation of the probability of saturating the use of computing resources for various SINR and number of tasks.

Figure 89 and Figure 90 show the probability of high saturation in the usage of communication and computing resources, respectively. This probability represents the likelihood that communication or computing resources are utilized beyond 80% of their total capacity when new tasks arrive. Results are presented for different combinations of the number of tasks and SINR values. The figures demonstrate that the deterministic proposal (Det in figures) manages communication and computing resources more effectively compared to the benchmark scheme (Min in figures), as it reduces the probability of resource saturation. This is particularly significant because higher saturation probabilities increase the risk that new tasks will lack the necessary communication or computing resources to meet their deadlines. The figures show that the risk of saturation decreases as the SINR improves or the task load decreases. It should also be highlighted that the standard deviation of the saturation probabilities is smaller under better SINR conditions. The highest variability is observed when the number of tasks is 45 or 55, as these workloads are close to the capacity limit (Figure 87).

Figure 91 Percentage of allocated tasks to computing units for various SINR values and number of tasks.

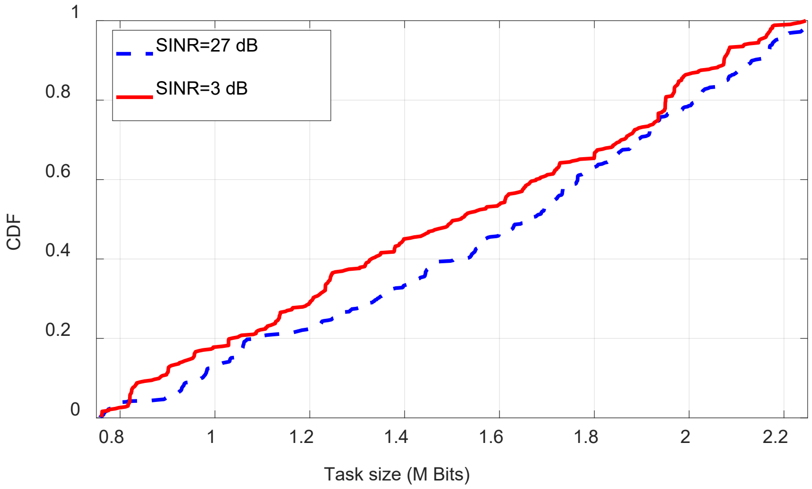


Figure 92 CDF of size of offloaded tasks to 6G network in different link qualities (SINR=3dB and SINR=27dB) for Deterministic Proposal.

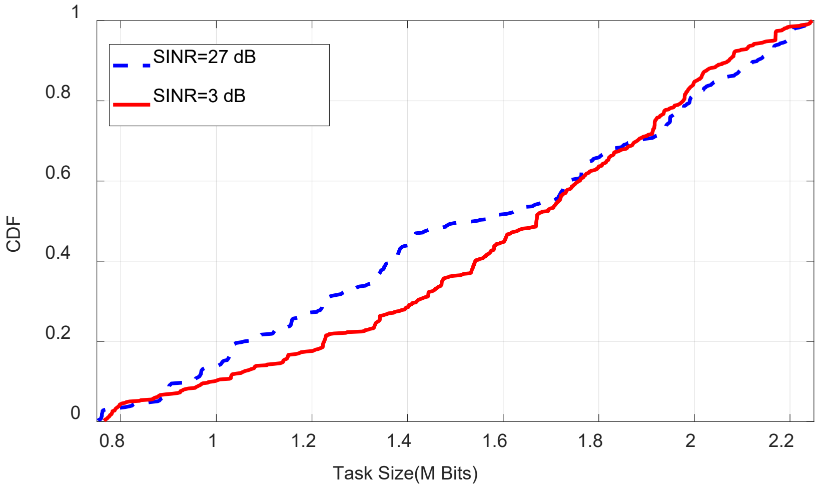


Figure 93 CDF of size of offloaded tasks to 6G network in different link qualities (SINR=3dB and SINR=27dB) for Minimum scheme.

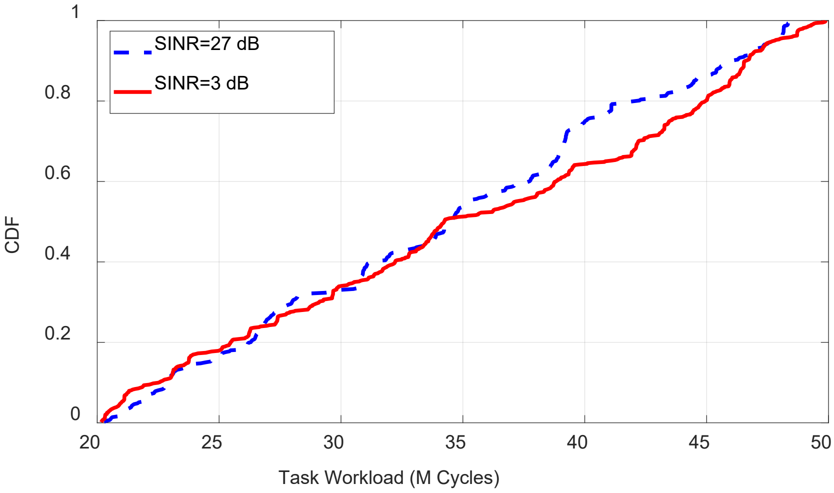


Figure 94 CDF of workload of offloaded tasks to 6G network in different link qualities (SINR=3dB and SINR=27dB) for Deterministic Proposal.

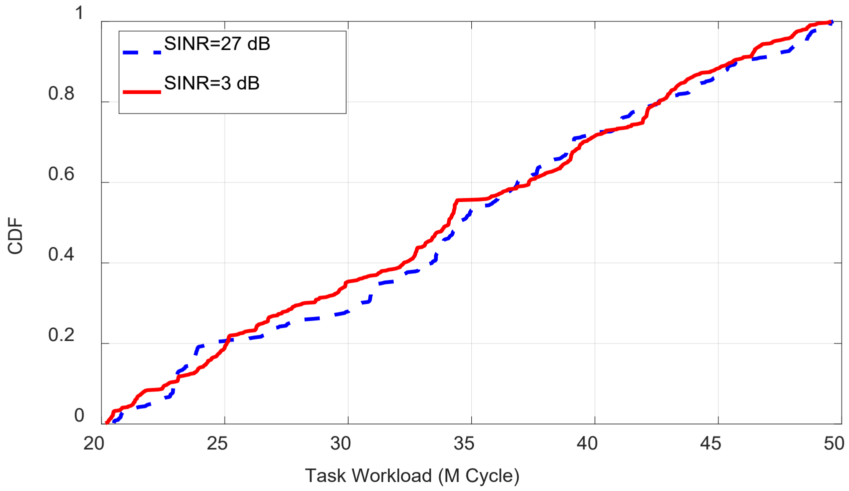


Figure 95 CDF of workload of offloaded tasks to 6G network in different link qualities (SINR=3dB and SINR=27dB) for Minimum scheme.

Figure 91 shows the percentage of tasks allocated to different computing units in the local/IoT-edge-cloud continuum under both good and bad SINR conditions for the connection to the 6G parent network, and for different task loads. The figure shows that when SINR = 3 dB, the deterministic proposal offloads fewer tasks to the edge and cloud compared to the benchmark scheme. This is because poor link quality conditions increase the risk of saturating communication resources, as more robust modulation and coding schemes required in such conditions reduce the achievable link data rates. In contrast, the benchmark scheme attempts to offload more tasks to the edge and cloud due to their higher processing power, which ultimately penalizes the system's ability to satisfactorily support more tasks (Figure 83 to Figure 86). On the other hand, when SINR = 27 dB, the deterministic scheme offloads more tasks to the edge and cloud compared to the benchmark scheme and to the scenario with the lower SINR, demonstrating its capacity to adapt its offloading and allocation decisions based on the operating conditions.

Figure 92 to Figure 95 compare the distribution of task size and workload offloaded to the parent network when the number of tasks is 45, which is close to the system's capacity. Figure 92 and Figure 93 show that the deterministic scheme offloads smaller tasks when the quality of the link to the network is low (SINR = 3dB), as it adapts to the reduced data rate caused by the low SINR. In contrast, the benchmark scheme offloads larger tasks under the same conditions. Figure 94 and Figure 95 show that the deterministic scheme offloads tasks with higher workloads to the 6G network when the link quality is poor, whereas the benchmark scheme shows no such preference. These results show that the deterministic scheme intelligently offloads tasks with higher workloads and smaller sizes under low SINR conditions. Consequently, the few tasks that can be offloaded to the network when the link data rate is low are those that will benefit the most from the higher processing power of the edge and cloud, maximizing resource utilization.

4.3.7 Scalable Deterministic Task Offloading and Resource Allocation

Realizing the potential of the IoT-edge-cloud continuum requires task offloading and resource allocation strategies that jointly manage and optimize communication and computing resources across the continuum [40]. Several strategies have been proposed to date, with a common primary focus for most of them on minimizing computational and communication latencies, which is particularly relevant for low latency applications. For example, Cai et al [50] analysed the trade-offs between local and remote task processing to minimize latency while considering bandwidth and processing power constraints. In [51], the authors propose a reverse offloading framework to reduce system latency by opportunistically utilizing resources either at the edge or at vehicles to process large amounts of data. Similarly, the proposal in [47] seeks to minimize processing delays by leveraging idle resources at devices and offloading tasks to these devices under increasing processing demands. In [52], Oliveira et al. propose a task allocation strategy that minimizes response times for latency-sensitive applications while reducing network traffic by mitigating idle resource time in hierarchical fog architectures. In contrast to minimizing latency or task execution time, we advocate for a deterministic approach to task offloading and resource allocation that prioritizes increasing the number of tasks executed within their deadlines over reducing task execution or completion time. Deterministic schemes can flexibly manage task completion deadlines to balance workload and resource distribution across the continuum, which in turn has the potential to enhance scalability by improving the system's ability to meet task execution deadlines for a larger number of tasks. To evaluate this potential, we compare the scalability of a deterministic task offloading and resource allocation scheme (referred to as Deterministic in this study) against a reference state-of-the-art strategy focused on reducing task completion time. Specifically, we compare the performance of a deterministic scheme against a reference scheme from [50] referred to as Minimum in this study, that we implement in our system model presented in Section 4.3.3 along with a random strategy. The following sub-sections describe the objective functions and common system constraints for these three strategies.

4.3.7.1 Objective functions

Minimum allocates communication and computing resources within the IoT-edge-cloud continuum with the primary objective of minimizing task execution time. Its objective function can be formulated as:

$$min \sum\nolimits_{n} \sum\nolimits_{i} T_{i,n}$$

where $T_{i,n}$, as defined in Section 4.4.3, represents the execution time of the task $f_{i,n}$ generated in the subnetwork n.

The Deterministic scheme is designed to ensure that tasks are executed before their deadlines (i.e. $T_{i,n} < T_{i,n}^{max}$) rather than focusing on minimizing execution time. The objective is for Deterministic to leverage the flexibility and varying deadlines of tasks to distribute and balance them across the IoT-edge-cloud continuum. Its objective function is formulated as:

$$min \sum\nolimits_{n} \sum\nolimits_{i} \beta\left(\frac{T_{i,n}}{T_{i,n}^{max}}\right),$$

where $T_{i,n}$ represents the execution time of task $f_{i,n}$ generated in the subnetwork n, $T_{i,n}^{max}$ is the task's deadline, and β is a penalty function defined as:

$$\beta(\xi) = \begin{cases} 0, & 0 \leq \xi \leq 1, \\ M, & \xi \geq 1. \end{cases}$$

In $\beta(\xi)$, ξ represents the normalized execution time $T_{i,n}$ of a task relative to its deadline $T_{i,n}^{max}$, and M is a large positive constant. The penalty function in $\beta(\xi)$ is introduced to ensure a deterministic execution of tasks within their bounded deadlines, as it penalizes tasks whose execution $T_{i,n}$ exceeds the deadline $T_{i,n}^{max}$.

A baseline third Random scheme randomly selects the computing unit to process the task $T_{i,n}$ generated in the subnetwork n. If the selected unit requires moving the task to another unit of the IoT-edge-cloud continuum, the communication resources needed to transmit the task to the selected unit are also chosen randomly from the available options. Nevertheless, the communication and computing resources are allocated to ensure that each task is processed within its deadline $T_{i,n}^{max}$.

4.3.7.2 Common constraints to all considered schemes

For fairness, all schemes (Minimum, Deterministic and Random) are defined with six common constraints in the task offloading and resource allocation processes.

The first constraint is the binary task allocation constraint, which states that task $f_{i,n}$ generated in subnetwork n is allocated to a single computing unit and cannot be divided among multiple units. The mathematical expression for this constraint is given as:

$$\sum\nolimits_{j=1}^{J} a_{i,n}^{(j)} + \sum\nolimits_{q=1}^{Q} a_{i,n}^{(q)} + a_{i,n}^{(c)} = 1, \quad \forall\, i\,, n,$$

where J is the number of local computing units in the subnetwork n (i.e., the sum of LC_n, and HC_n units) and Q is the number of edge computing units. $a_{i,n}^{(j)}$, $a_{i,n}^{(q)}$ and $a_{i,n}^{(c)}$ are binary variables equal to 1 if task $f_{i,n}$ generated in subnetwork n is allocated to the local computing unit j, the edge unit q, or the cloud unit c, respectively, and 0 otherwise.

In accordance with the principles of OFDMA, the second constraint ensures that communication resources assigned to subnetworks (i.e., $k_s \epsilon K_s$) can be utilized by only one communication link l at a time, preventing transmissions collisions within the subnetwork. The mathematical formulation of this constraint is expressed as:

$$\sum\nolimits_{l} \sum\nolimits_{i} b_{l,i,n}^{(k_s)} \leq 1, \qquad \forall\, k_s, n,$$

where $b_{l,i,n}^{(k_s)}$ is a binary variable equal to 1 when communication resource k_s is allocated to transmit task $f_{i,n}$ generated in the subnetwork n to the link l.

The third constraint ensures that the shared communication resources for subnetworks to connect to the wide-area cellular network (i.e., $k_p \epsilon K_p$) can be allocated to only one communication link at a time. Hence, this constraint prevents transmissions collisions between subnetworks accessing the wide-area cellular network. The mathematical expression for this constraint is given as:

$$\sum\nolimits_{n} \sum\nolimits_{l} \sum\nolimits_{i} b_{l,i,n}^{(k_p)} \leq 1\,, \qquad \forall\, k_p,$$

where $b_{l,i,n}^{(k_p)}$ is a binary variable equal to 1 when communication resource k_p is allocated to transmit task $f_{i,n}$ in link l between the subnetwork n and the wide-area network.

The fourth constraint ensures that the total transmission rate of all tasks utilizing links between subnetworks and the wide-area network does not exceed the maximum achievable data rate of that link. The mathematical expression of this constraint is:

$$\sum\nolimits_{i} r_{l,i,n}(t) \leq r_{l,n}(t), \;\; \forall l, n,$$

where $r_{l,i,n}(t)$ is the data rate of link l for transmitting task $f_{i,n}$ generated in subnetwork n and $r_{l,n}(t)$ is the maximum possible data rate of link l.

The fifth and sixth constraints ensure that the total processing workload of tasks assigned to a local computing unit within subnetwork n (fifth constraint) or to an edge and cloud computing units (sixth constraint) does not exceed the maximum processing capacity of that unit over a given time interval. The mathematical formulation of these constraints is given as:

$$\sum\nolimits_{i} c_{i,n} a_{i,n}^{(x_s)} \leq C_{x_s,n}^{max}, \qquad \forall\, x_s, n,$$

$$\sum\nolimits_{i} \sum\nolimits_{n} c_{i,n} a_{i,n}^{(x_2)} \leq C_{x_p}^{max}, \qquad \forall\, x_p,$$

where $C_{x_s,n}^{max}$is the maximum processing capacity of computing unit x_s in subnetwork n, and $C_{x_p}^{max}$is the maximum processing capacity of computing unit x_p (edge or cloud nodes).

4.3.8 Evaluation of Scalability of Deterministic Scheme

4.3.8.1 Scenario

Without loss of generality, we evaluate the scalability of the task offloading and resource allocation schemes in a 6G-envisioned industrial cyber-physical scenario in which wireless subnetworks formed by mobile robots are connected with a cellular network [53]. This scenario enables the processing of the tasks generated by the robots in the IoT-edge-cloud continuum. We evaluate scenarios involving varying numbers of subnetworks (N), ranging from 2 to 5, and different numbers of tasks (I), ranging from 5 to 25. We consider each subnetwork includes 15 SNEs representing robots' sensors, and 4 LCs and 1 HC computing units for processing, monitoring, and control tasks within the subnetwork. While tasks can be generated by any element in the subnetwork, local processing within subnetworks is limited to LCs and HCs. The processing power of {LC, HC, edge, and cloud} is {2.5, 5, 70 and 150} GHz following [46]. Within each subnetwork, {60%, 20%, and 20%} of tasks are generated by {SNEs, LCs, and HC}, respectively. We consider tasks processing workloads and sizes to emulate the varying requirements of traffic generated from mobile robots. Following [54], we consider that mobile robots tasks require between 20 and 50 Mcycles and range in size from 0.75 to 2.25 Mbits, with an average size of 1 Mbits. The processed result size for each task is set to 15% of its original size. Following the requirements for cyber-physical control of mobile robots defined in [55], task deadlines (T_i^{max}) are randomly allocated within the 20 to 100 ms range. The penalty value M assigned to tasks that are not completed by their deadlines is 100. Subnetworks are configured to operate with a dedicated bandwidth BW_s of 100 MHz, while the links between subnetworks and the wide-area cellular network use a bandwidth BW_p of 50 MHz. Following [42], we assume that wireless links within subnetworks maintain an average SINR of 30 dB, with the channel fading modelled with a Rayleigh distribution. The average SINR for the links between subnetworks and the wide-area cellular networks varies between 0 dB and 30 dB. We consider a subcarrier spacing (SCS) of 30 kHz and a time slot duration of 0.5 ms [43].

We have implemented a genetic algorithm to solve the optimization problems of the task offloading and resource allocation schemes. Considering the number of variables and possible resource allocation options in the evaluated scenarios, the optimization process starts with a population of a thousand candidate resource allocation solutions. The top 20% of the best-performing candidates are retained for the next generation, while the remaining 80% are created through crossover operations from the top 20%. The algorithm iterates over ten generations. A mutation rate of 20% introduces small random changes to enhance diversity and prevent premature convergence. The configuration of the algorithm balances performance and computation complexity, and our tests showed that it converged to near-optimal solutions [45].

4.3.8.2 Results

Figure 96 and Figure 97 depict the average ratio of satisfied tasks as a function of the number of executed tasks per subnetwork and number of subnetworks, under good and poor SINR conditions in the connection to the wide-area cellular network, respectively. The satisfaction ratio represents the proportion of tasks completed before their deadlines relative to the total number of tasks and is averaged across all subnetworks. The obtained results show that Deterministic achieves the highest satisfaction ratio, regardless of the number of tasks, the number of subnetworks, or the connection quality to the wide-area network. For example, the obtained results show that the Deterministic scheme supports all 25 tasks generated across the 5 different subnetworks, with each subnetwork generating 5 tasks. In contrast, the number of satisfied tasks is reduced by an average of {10%, 38%} for SINR=30dB (Figure 96) and {18%, 45%} for SINR=0dB (Figure 97) for the Minimum and Random schemes, respectively. Increasing the number of tasks per subnetwork reduces the satisfaction ratio for all schemes due to the limited computing and communication resources available in the scenario compared to the simulated workloads. However, the reduction observed is less significant for the Deterministic scheme, which highlights its better scalability. This is also observed as the number of subnetworks augments. In this case, the satisfaction ratio does not decrease with the Deterministic, while the satisfaction ratio for Minimum and Random decreases as the number of subnetworks increase. These results emphasize the scalability advantage of deterministic policies that jointly allocate and manage communication and computing resources across the IoT-edge-cloud continuum with the objective of satisfying the tasks' execution deadlines rather than minimizing tasks' execution time. Trying to minimize each task's execution time can result in many tasks trying to use computing units with higher computing capabilities or links with high data rates, increasing the pressure on these resources. The results obtained show that this can ultimately lead to the overload of these resources and a lower satisfaction ratio and poorer scalability compared to deterministic policies. This is particularly visible under bad link quality conditions (Figure 97) as Minimum can result in satisfaction ratios even lower than with Random for the highest number of subnetworks and of tasks per subnetwork analysed. The Deterministic scheme focuses on ensuring all tasks are executed before their deadlines rather than minimizing their execution time. Deterministic leverages the flexibility and varying deadlines of tasks to distribute and balance tasks across the continuum. By balancing the workload across the continuum, Deterministic avoids putting excessive strain on certain computing and communication resources, thus improving overall scalability.

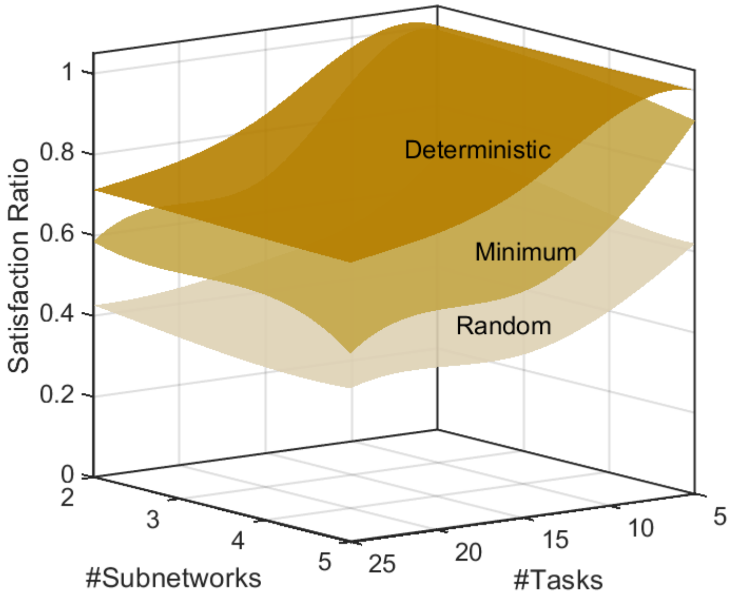


Figure 96 Average ratio of satisfied tasks as a function of number of subnetworks and tasks per subnetworks when average SINR=30 dB.

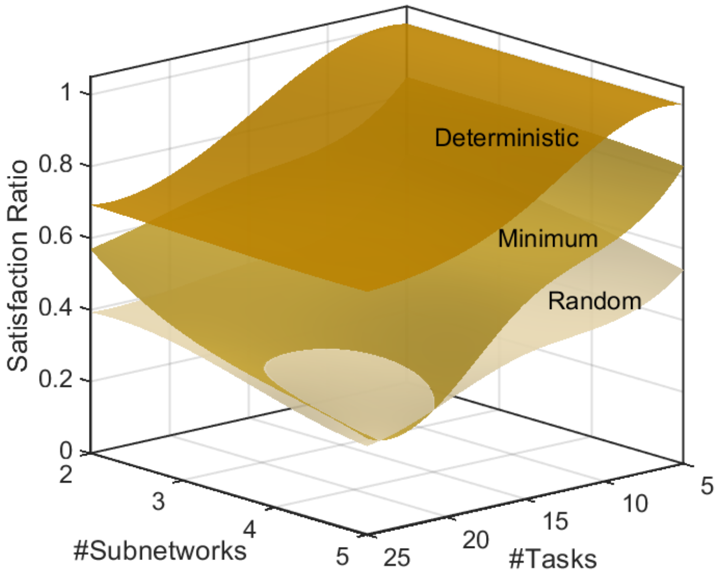
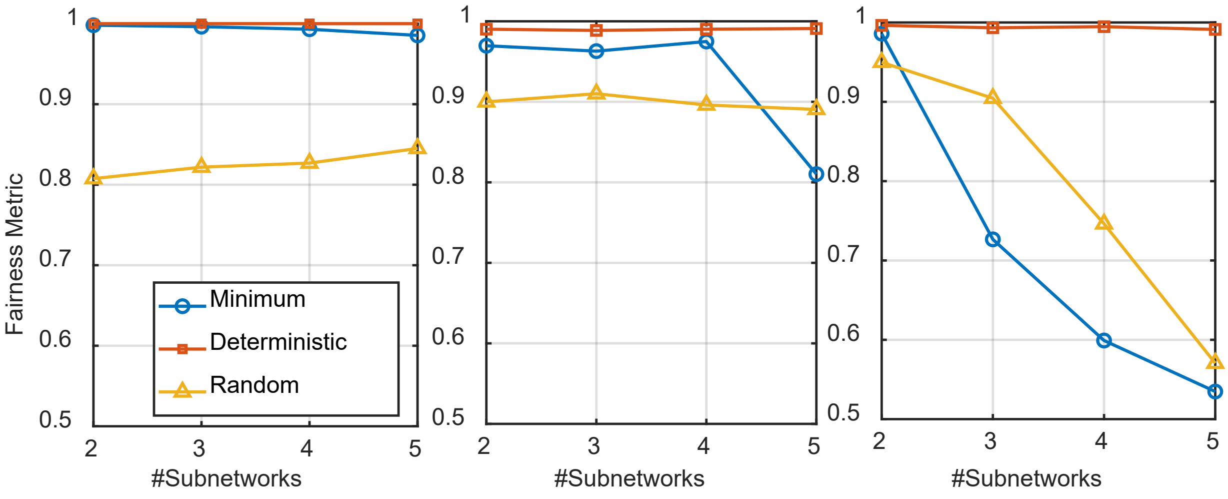


Figure 97 Average ratio of satisfied tasks as a function of number of subnetworks and tasks per subnetworks when average SINR=0 dB.

The scalability of the task allocation schemes also depends on the fairness in task satisfaction across subnetworks. The average satisfaction ratio shown in Figure 96 and Figure 97 does not reflect the balance in task satisfaction across subnetworks Figure 98 reports the Jain Fairness Index (JFI) metric to measure the distribution of task satisfaction ratios among different subnetworks **[56]**. The JFI for satisfaction ratio of a set of subnetworks ($SR_n, n \,\epsilon \{1, \dots, \ N\}$) can be expressed as:

$$JFI(SR_n) = \frac{\left(\sum_n SR_n\right)^2}{N \sum_n SR_n^2}.$$

The JFI takes values ranging from 0 to 1. A value close to 1 signifies a more equitable distribution of the satisfaction ratio among different subnetworks, while a value closer to 0 indicates significant disparities. Figure 98 reports the measured JFI for scenarios with 2 to 5 subnetworks and an increasing number of tasks per subnetwork: 5 in Figure 98.a, 15 in Figure 98.b and 25 in Figure 98.c. The obtained results demonstrate that Deterministic consistently achieves a JFI value close to 1 across all scenarios, regardless of the number of tasks or subnetworks. This indicates a high level of fairness in the distribution of satisfied tasks among different subnetworks. In contrast, the JFI values for Minimum and Random decrease as the number of subnetworks and tasks increases, which highlights lower fairness and significant variations in the number of satisfied tasks across subnetworks.

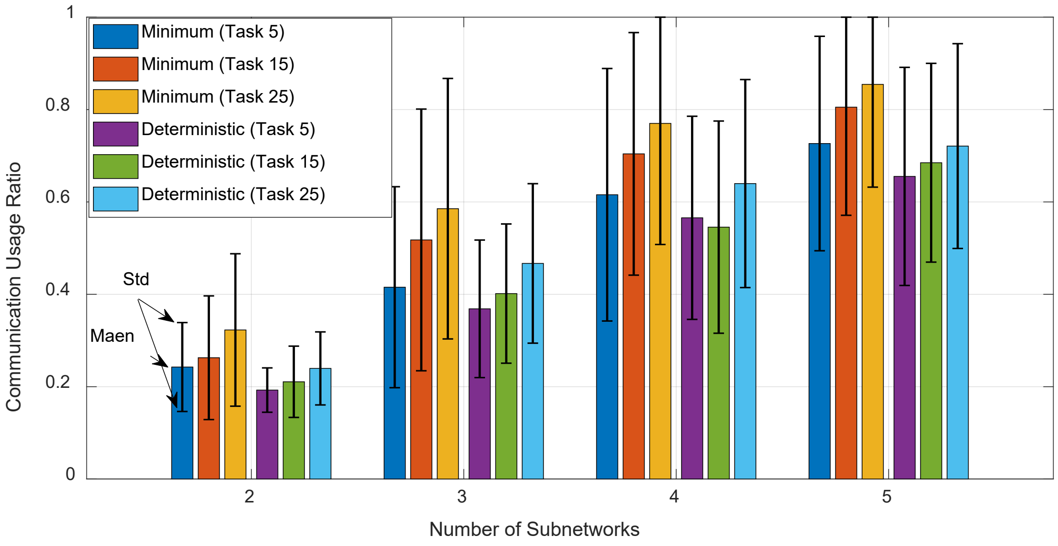


a) *N*=5 b) *N*=15 c) *N*=25

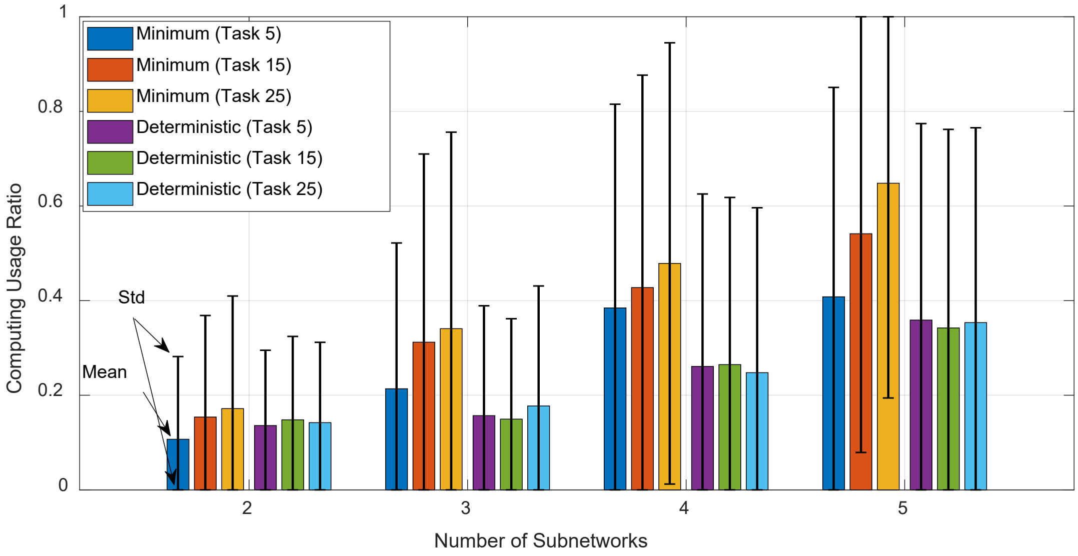
Figure 98 Jain Fairness Index (JFI) for the distribution of satisfied tasks for average SINR = 30 dB (similar trends observed for SINR = 0 dB).

The scalability benefits of Deterministic over the other evaluated schemes stem from its better distribution and balance of tasks and workload across the IoT-edge-cloud continuum, achieved by leveraging the varying deadlines of tasks. This is illustrated in Figure 99 which depicts the average ratio of utilized communication and computing resources of the link(s) and processing unit selected by the task offloading and resource allocation schemes1 relative to the total number of resources. The utilization ratio is measured from the moment the task is allocated to the selected link(s) or processing unit until its deadline. The bars indicate the average resource utilization ratio, while the lines within each bar represent the standard deviation. The obtained results show that Deterministic selects less congested communication and computing resources (i.e. with a lower utilization ratio) compared to Minimum. For example, Deterministic selects communication links and processing units that are up to 17.5% (Figure 99.a) and 45% (Figure 99.b) less utilized – based on average values - than those selected by Minimum, respectively. The standard deviation values also show that Deterministic results in resource utilization ratios below 100% even in the scenarios with the largest number of subnetworks and task loads. In contrast, in scenarios with 4 and 5 subnetworks and 15 and 25 tasks per subnetwork, Minimum causes bottlenecks by reaching full resource utilization (100%), leading to system congestion, and potentially preventing certain tasks from meeting their deadlines.

1.The previous analysis has shown that *Random* is outperformed by the other schemes and is omitted from the figure for clarity.



a) Communication resources utilization ratio of the selected link(s).



b) Computing resources utilization ratio of the selected processing unit.

Figure 99 Average and standard deviation of resources utilization ratio when average SINR=30 dB (similar trends observed for average SINR = 0 dB).

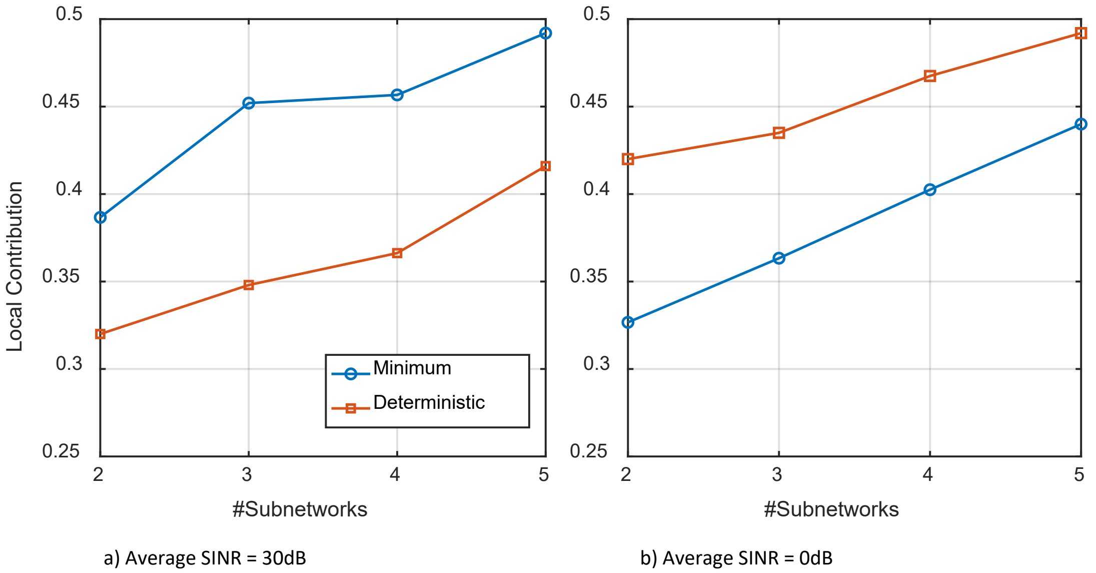


Figure 100 Average ratio of tasks processed at subnetworks when the number of tasks per subnetwork is 15 (similar trends were observed for other values).

By balancing the workload across the continuum (Figure 99), Deterministic avoids putting excessive strain on the network's computing and communication resources, thus improving overall scalability. Figure 99 shows that, on average, communication resources experience higher utilization than computing resources as the number of subnetworks and tasks increases. This is the case because all subnetworks share the same spectrum to connect to the wide-area cellular network and reach the edge node and cloud server. Figure 100 shows the average ratio of tasks processed locally (within the subnetworks) by Minimum and Deterministic. Both schemes increase the ratio of tasks processed locally, i.e., within the subnetworks, as the number of subnetworks and total number of tasks grows. This is due to the scarcity of communication resources that all subnetworks must share to connect to the cellular network. However, Figure 100 reveals that, compared to Minimum, Deterministic processes a higher proportion of tasks locally when the quality of the link from the subnetwork to the wide-area cellular network is poor (Figure 100.b), whereas it relies less on local processing when the link quality is good (Figure 100.a). These results reveal the ability of Deterministic to adapt the offloading of tasks based on the condition of shared communication resources, ensuring efficient and balanced task distribution across the continuum while augmenting the number of tasks that are executed before their deadline.

4.3.9 Summary

This study has demonstrated that a deterministic task offloading and resource allocation scheme for the joint management of communication and computing resources in the IoT-edge-cloud continuum enhances scalability in next-generation cellular networks. The proposed scheme prioritizes meeting task deadlines over simply minimizing individual task execution latency, allowing for more efficient and balanced workload distribution across the continuum. By flexibly managing task completion deadlines, this approach optimizes resource utilization, prevents resource saturation, and avoids system bottlenecks, making the network more resilient to varying computing workloads and communication quality conditions.

This study also has demonstrated that a deterministic approach to task offloading and (communication and computing) resource allocation in the IoT-edge-cloud continuum can enhance scalability in next-generation cellular networks. By flexibly managing task completion deadlines, a deterministic strategy achieves a more balanced workload and resource distribution across the continuum than existing methods and can better adapt to varying operating conditions (e.g., link quality). This augments task satisfaction ratios and fairness across the system while enabling more efficient resource utilization, which helps prevent resource saturation and enhances scalability.

4.4 FLEXIBLE LOCAL ROUTING IN SUBNETWORKS FOR TASK OFFLOADING

4.4.1 Introduction

The rapid automotive evolution has led to an increasing demand for complex and diverse in-vehicle functions, driven by industry trends such as connectivity, electrification, automated driving, and smart mobility. This evolution needs increasingly sophisticated in-vehicle computing capabilities to support features and services with stringent reliability and deterministic service level requirements. Additionally, the computational and operational demands of next-generation automotive systems require evolving from traditional in-vehicle electrical/electronic (E/E) architectures with distributed processing and domain-specific controllers [57]. To address these growing demands, the automotive industry has been shifting towards software-defined vehicles (SDVs), which enable more flexible, configurable, scalable, and upgradable in-vehicle functionalities [58].

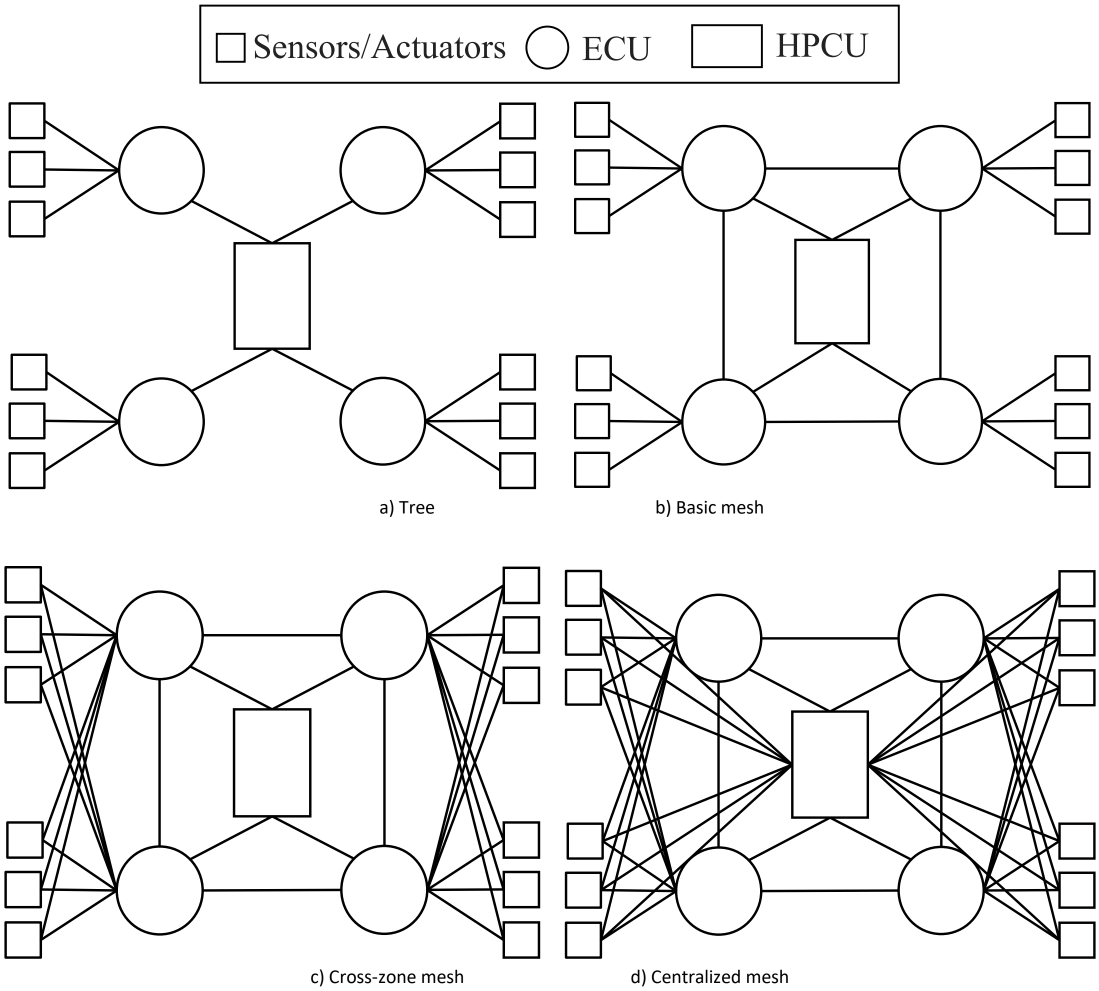
The transition to SDVs necessitates significant changes in the in-vehicle network (IVN) and E/E architecture. Traditional architectures, which rely on numerous independent electronic control units (ECUs) to manage sensors and actuators for specific vehicle subsystems, are giving way to zonal IVN architectures with centralized computing [59]. In this new architecture, the vehicle's computational workload is handled by high-performance central compute platforms and zonal controllers, enabling improved coordination between vehicle functions and data fusion, thereby facilitating higher levels of automation and intelligence in decision-making. The zonal IVN architecture with centralized computing also incorporates redundant, diverse, and fault-tolerant elements and links, which are essential for achieving the functional safety levels required in critical systems. However, the efficiency of this IVN architecture depends on the effective management and scheduling of computational tasks, making task scheduling a critical requirement. As vehicle functionalities expand, optimizing workload distribution across processing units is essential to maintain deterministic responsiveness and dependability of critical functions. Traditional static scheduling approaches, where tasks are allocated to predefined computing units or ECUs, may face challenges in this dynamic and demanding environment. The automotive industry advocates for the use of global scheduling with adaptive task allocation approaches to effectively balance workloads across available communication links and computational resources without compromising the execution of critical vehicle functions [57].

In this context, this study advances the state of the art with the proposal of a deterministic task scheduling approach for zonal in-vehicle E/E architectures with centralized computing. The study demonstrates that a deterministic task scheduling can better guarantee the deterministic service levels of critical in-vehicle functions than alternative approaches that schedule tasks based on the shortest path or the objective to minimize task execution time. Our evaluation also demonstrates that a deterministic task scheduling can satisfactorily support increasing in-vehicle computational workloads and tasks and achieve a more balanced workload and resource utilization across the zonal in-vehicle network. We demonstrate that the benefits achieved with a deterministic task scheduling approach are valid across a variety of IVN topologies, ranging from traditional tree-based topologies to mesh topologies with centralized computing based on realistic case studies [60][61]. These benefits are also maintained considering hybrid wireless-wired IVN implementations, where a gradual introduction of wireless offers increased connectivity diversity for linking sensors and actuators to computing units. The results demonstrate that the deterministic task scheduling approach can better adapt to varying operating conditions while enabling efficient resource utilization, thereby preventing resource saturation, and enhancing scalability.

4.4.2 In-Vehicle Networks and Topologies

Traditional IVN architectures incorporate one ECU for each in-vehicle electronic function with a very specific control task, and a direct interconnection among them. This approach requires new ECUs and interconnections when new sensors or actuators are required. The significant increase in electronic functions has triggered an evolution of IVN architectures for better scalability. A first evolution has been domain-based IVN architectures with several functional domains (infotainment, powertrain, assisted driving, etc.) managed by domain-specific networking technologies and controllers. Despite its benefits, domain-based IVN architectures experience challenges for developing automotive applications that require cross-domain functionality, a need that is growing with vehicle softwarization and the gradual introduction of autonomous driving functions. Zonal IVN architectures have emerged as an alternative to enhance efficiency as vehicle complexity and functionality increase. Zonal IVN architectures group embedded devices and electronics based on physical location rather than logically or per domain. The zonal IVN architecture locally connects sensors and actuators to zonal controllers or ECUs that are physically and strategically distributed through the vehicle. These zonal controllers rely on a high-speed backbone network to connect to each other and to the vehicle's high-performance central computing platform with advanced processing capabilities. A trend in the evolution of zonal IVN architectures is vehicle-centralized computing [59], and the possibility that sensors/actuators may bypass the zonal ECUs and connect directly to the vehicle's central computing platform.

In line with the transition to zonal IVN architectures with centralized computing, this study analyses four in-vehicle network topologies, depicted in Figure 101, which are based on realistic case studies from [60][61]. The topologies share a common structure, defining four in-vehicle zones that represent the front-left, front-right, rear-left and rear-right areas of the vehicle. Each zone includes a zone ECU in addition to the sensors and actuators located within that area. The topologies also incorporate a central High-Performance Computing Unit (HPCU). However, they differ in their degree of connectivity. Figure 101.a represents a conventional tree IVN topology, where sensors and actuators are connected to their respective zonal ECUs, and the zonal ECUs are connected to the central HPCU. Figure 101.b represents a basic mesh IVN topology which introduces a connectivity backbone between zone ECUs. Figure 101.c follows the IVN topology of the Orion Crew Exploration Vehicle (CEV) as utilized in [60], and we refer to it as cross-zone mesh. This topology adds cross-zone connections, providing redundant links between sensors/actuators and nearby zone ECUs. Without loss of generality, we consider these cross-zone connections link front and rear sensors/actuators to the zone ECUs located in the opposite area (i.e., left to right and right to left). Finally, Figure 101.d depicts a centralized mesh topology, which introduces direct links between the sensors/actuators and the HPCU to the cross-zone mesh topology.



a) Tree

b) Basic mesh

c) Cross-zone mesh

d) Centralized mesh

Figure 101 In-vehicle network topologies.

4.4.3 System Model

The system consists of automotive in-vehicle functions that generate tasks f_n, where $n\epsilon\{1, \dots, N\}$). Tasks can be originated from sensors and actuators (SNA_{s_m}), zone ECUs ($zECU_m$), and the HPCU (H), where $m\epsilon\{1, \dots M\}$ represents the in-vehicle area or zone (M=4), and $s_m\epsilon\{1, \dots S_m\}$ is the number of sensors/actuators in the zone m. Each task f_n is characterized by the tuple (c_n, s_n, s'_n, t_n, T_n^{max}) where: c_n denotes the computing demand of the task, s_n represents the task's size, s'_n is the size of the task after processing, t_n indicates the task generation time, and T_n^{max} defines the task deadline for processing. The processing of tasks is restricted to $zECU_m$ and H. We consider that task scheduling schemes (described in Section 4.4.4) dynamically assign tasks to computing units within the IVN. When a task f_n is executed on a processing unit different from where it was generated, the processed result with size s'_n must be transmitted back to its source unit. The T_n^{max} of task f_n accounts then for the transmission time to move the task to the assigned processing unit, the processing duration, and the time required to transmit the processed result back to its source unit.

The in-vehicle computing units have different processing power, denoted by P_x, and a maximum processing capacity C_x^{max} over a time period T with $C_x^{max} = P_x \cdot T$, where $x \; \epsilon\{z, h\}$ refers to the type of processing unit, i.e., $\{zECU_m, H\}$, respectively. The time required to process a task f_n on a computing unit $x\epsilon\{z, h\}$ is given by:

$$t_p^n = \frac{c_n}{P_x}.$$

We consider that the IVN topologies depicted in Figure 101 can be fully wired or hybrid wireless-wired. In both cases, we represent with E the set of links between the set of the IVN elements (SNA_{s_m}, $zECU_m$, H). d_{ij} represents the distance between nodes i and j in the IVN. We consider that nodes that can be reached directly are closer than those that require passing through other nodes.

For wired links, we consider Ethernet-like connections with a data rate R_w and no transmissions errors (i.e., the reliability ρ is equal to 1), ensuring reliable and consistent data transmission. The transmission time over a wired link $w \epsilon E$ can then be computed as:

$$t_c^n = \frac{s_n}{R_W}.$$

In the hybrid wireless-wired IVN scenarios, we restrict the use of wireless connectivity to links between sensors and actuators and the IVN units to which they can connect. The wireless connections therefore depend on the topology (see Figure 101). Wireless links are prone to transmission errors and are characterized by a reliability ρ < 1. We consider that the wireless links utilize an Orthogonal Frequency Division Multiple Access (OFDMA)-based radio access interface. A dedicated band with bandwidth BW_m is assigned for each of the 4 in-vehicle zones. Additionally, communication between sensors/actuators of all zones and the HPCU in the centralized mesh IVN topology (Figure 101.d) uses a dedicated band of bandwidth BW_h. Each BW_x ($x \epsilon \{z, h\}$) is divided into K_x orthogonal resources. Then, the data rate available at any given time for communication resource $k\epsilon\{K_x\}$ in the wireless link $l \epsilon E$ is denoted as $r_l^{(k)}(t)$:

$$r_l^{(k)}(t) = BW_k \cdot log_2\big(1 + \gamma_l(t)\big)(1 - BER),$$

where BW_K represents the bandwidth of the communication resource k,$\gamma_l(t)$ denotes the Signal-to-Interference plus Noise Ratio (SINR) at time t of the wireless link l, and BER is the bit error rate, which depends on the modulation and coding scheme employed in the communication resource k. To model channel fading effects, we assume a Rayleigh distribution. The total data rate of the link l is calculated as the sum of the data rates for all communication resources k utilized in the link:

$$R_l(t) = \sum\nolimits_k r_l^{(k)}(t).$$

The transmission time over the wireless link l is then:

$$t_c^n = \frac{s_n}{R_l(t)}.$$

Similarly, the transmission time over wireless communication links for the processed result of a task f_n with size of s'_n can be expressed as t'^n_c and is computed following t_c^n using s'_n instead of s_n.

The total execution time T_n required to complete a task f_n includes the communication time to transmit the task to the processing unit (t_c^n), the processing time at the computing unit (t_p^n), and the communication time to return the processed result (t'^n_c). The total execution time is given by:

$$T_n = t_c^n + t_p^n + t'^n_c\ .$$

4.4.4 Deterministic Task Scheduling

This study proposes a deterministic task scheduling scheme for IVNs. The Deterministic scheme prioritizes maximizing the number of tasks completed within their deadlines (i.e., $T_n \leq T_n^{max}$), making it particularly suitable for guaranteeing the timely execution of critical vehicle functions within strict bounded time constraints. The scheme dynamically adjusts task completion times based on varying deadlines, enabling flexible management and balanced workload distribution across the IVN. The objective function is formulated as:

$$min \sum\nolimits_n \beta\left(\frac{T_n}{T_n^{max}}\right),$$

where $\beta(\xi)$ is a penalty function defined as:

$$\beta(\xi) = \begin{cases} 1 - \prod_{(i,j)\in E_n} x_{ij}.\rho_{ij}, & 0 \leq \xi \leq 1, \\ 1, & \xi \geq 1, \end{cases}$$

where x_{ij} is a binary decision variable which is equal to 1 if the task is routed through the link between node i and node j, and ρ_{ij} is the reliability of the link between node i and node j. By introducing the reliability of the IVN links in $\beta(\xi)$, this scheme seeks selecting the most reliable route possible from the multiple available paths when allocating the task from the source to the computing unit. $\beta(\xi)$ also ensures that if a task exceeds its deadline, a penalty of 1 is imposed, discouraging deadline violations.

4.4.4.1 Constraints

A first binary task scheduling constraint is defined as follows to ensure that task f_n is assigned to a single computing unit and cannot be split among multiple units:

$$\sum_{i=1}^{M+1} a_n^{(i)} = 1, \quad \forall n,$$

where M+1 represent the total number of computing units (i.e., M ECUs and 1 HPCU), and $a_n^{(i)}$is a binary variable equal to 1 if task f_n is allocated to the computing unit i.

The second constraint is only applicable to wireless links of the hybrid wireless-wired IVN topologies. Following OFDMA principles, this constraint ensures that communication resources from each band are allocated to only one communication link at a time. This prevents transmission collisions and enables interference-free communication between different zones and the HPCU.

$$\sum_{n=1}^{N} b_{l,n,z}^{(k)} = 1, \qquad \forall k, l, z,$$

where $b_{l,n,m}^{(k)}$ is a binary variable equal to 1 when communication resource k is allocated to transmit task f_n in link l of band BW_z ($z \in \{m, h\}$).

The third constraint ensures that the transmission rate for all tasks sharing a link does not exceed the link's maximum achievable data rate.

$$\sum_{n=1}^{N} R_{E,n}(t) \leq R_E(t), \;\; \forall l,$$

where $R_{E,n}(t)$ is the data rate of link $E = l \cup w$ for task f_n, and $R_E(t)$ is the maximum possible data rate of link E.

Finally, the fourth constraint ensures that the total processing workload of different tasks allocated to a computing unit within a specific time interval does not exceed the unit's maximum processing capacity.

$$\sum_{n=1}^{N} c_n a_n^{(x)} \leq C_x^{max}, \qquad \forall x,$$

where c_n denotes the computing demand of the task f_n, and C_x^{max} is the maximum processing capacity of unit $x \in \{z, h\}$.

4.4.4.2 Benchmark Schemes

The Deterministic proposal is compared against three benchmark schemes. The Baseline task scheduling scheme follows a traditional static approach, where tasks are allocated to predefined computing units [57]. For the IVN topologies defined in Section 4.4.2, this means that tasks generated by sensors/actuators are allocated to the ECUs within the same zone, while the ECUs and HPCU process their own tasks.

The Shortest task scheduling scheme follows the classic shortest-path approach [60][62] and focuses on minimizing the physical distance between the task source unit and the computing unit. Its objective function is formulated as:

$$\min \sum_{(i,j)\in E} d_{ij} \cdot x_{ij},$$

where d_{ij} represents the distance between node i and node j. x_{ij} is a binary decision variable equal to 1 if the task is routed through the link between node i and node j, and equal to 0 otherwise. $E = l \cup w$ is the set of IVN links.

The Minimum task scheduling scheme follows a common strategy used in task offloading processes [50]. Its objective is to allocate communication resources and computing units to minimize task execution time. This scheme transmits tasks by selecting jointly the fastest available path based on network topology and communication resources and fastest computing unit according to available processing capacity. The optimization function for this strategy is:

$$min \sum\nolimits_{n} T_n,$$

where T_n is defined in Section 4.4.3.

For fairness, the Shortest and Minimum schemes are defined with the same four constraints as Deterministic. Only the first three constrains apply for the Baseline scheme since it follows a predefined assignment of the computing units.

4.4.5 Evaluation Scenario

We analyse the impact of task scheduling on the performance of the IVNs following the topologies described in Section 4.4.2. The IVN consists of 36 sensors/actuators equally distributed in the 4 areas of the vehicle. Each area is controlled by a zone ECU, and central computing is performed in the HPCU. While tasks can be generated by any element of the IVN, only ECUs and the HPCU can handle processing. The processing power of the ECUs and the HPCU is set to 1 GHz and 4 GHz, respectively, based on the existing capabilities of off-the-shelf IVN processing units [63]. Within the vehicle, 70% of tasks are generated by sensors, 15% by ECUs, and 15% by the HPCU. We consider task processing workloads and sizes following the characterization of in-vehicle functions in [64]. In particular, we consider that tasks require between 5 and 15 Mcycles with an average of 10 Mcycles, and their size ranges between 0.5 and 1.5 Mbits, with an average size of 1 Mbits. The size of the processed result for each task is set to 15% of its original size. According to the requirements for in-vehicle functions identified in [61], task deadlines (T_n^{max}) are randomly assigned within the 40 to 100 ms range.

When hybrid wireless-wired IVN topologies are considered, the dedicated total bandwidth BW for wireless in-vehicle communication is 100 MHz, divided into five segments: each zone is assigned a bandwidth BW_m of 20 MHz, and the wireless connection to the HPCU has a dedicated bandwidthBW_h of 20 MHz. OFDMA communications are configured with a subcarrier spacing (SCS) of 30 kHz and a time slot duration of 0.5 ms following 3GPP TS 36.211 [43]. Based on empirical in-vehicle wireless measurements in [65], we assume that wireless links within the vehicle maintain an average SINR of 30 dB, with channel fading modelled using a Rayleigh distribution. The reliability ρ is considered to randomly vary in the range (0.95 – 1) for the wireless links between the sensors/actuators and their zonal ECU, and in the range (0.90 – 1) for the connections to cross-zone ECUs and the HPCU due to the largest distances and presence of blocking elements [65]. The wired links are modelled with an Ethernet-based data rate of 1 Gbps and ρ=1.

We implement a genetic algorithm to solve the NP-hard optimization problems of task scheduling as in [61]. The algorithm starts with 1,000 candidate solutions, retaining the top 20% for the next generation while generating the remaining 80% through crossover. Over ten generations, a 20% mutation rate introduces random variations to enhance diversity and prevent premature convergence. This configuration balances performance and computational complexity, achieving near-optimal solutions.

4.4.6 Evaluation Results

We first evaluate the ability of the task scheduling schemes under evaluation to successfully support in-vehicle tasks across different IVN topologies. A task is considered successfully supported if it is executed before its deadline. Figure 102 depicts the average satisfaction ratio as a function of the number of generated tasks for the fully wired implementation of the four IVN topologies. The satisfaction ratio represents the proportion of tasks completed before their deadlines relative to the total number of tasks. Note that non-satisfied tasks are also completed, but after their deadlines have passed. The results show that the Baseline scheme, which relies on pre-assignment of tasks to computing units, can support all generated tasks in scenarios with up to 25 tasks independently of the IVN topology. Its performance significantly degrades with higher workloads. A similar trend is observed for the Shortest task scheduling scheme even if it can dynamically schedule tasks across the IVN. This is the case because it does so considering only the physical topology of the IVN to find the shortest path and does not account for the computing capabilities and workloads of the units or the status of the links in the IVN. The Minimum task scheduling scheme does take into account this information to schedule tasks across the IVN to minimize task execution time. This approach outperforms the Baseline and Shortest schemes across all IVN topologies and achieves a satisfaction ratio above 95% in scenarios with up to 35 tasks. However, like the Baseline scheme, it can only fully execute all tasks within their deadlines in scenarios with up to 25 tasks. On the other hand, Figure 102 shows that the Deterministic scheme can fully satisfy a higher workload and achieves a satisfaction ratio above 95% in scenarios with up to 45 tasks (50 tasks in the centralized mesh). Under this load, the Deterministic scheme increases the ratio of satisfied tasks by {26.8%, 27.1%, 8.8%}, {27%, 27.4%, 9%}, {29.7%, 30%, 6.5%} and {30.8%, 30.9%, 6.3%} compared to the {Baseline, Shortest, Minimum} schemes for the tree-based, basic mesh, cross-zone mesh and centralized mesh topologies, respectively. These results clearly demonstrate that a deterministic task scheduling approach can better guarantee deterministic service levels and can support increasing in-vehicle computational workloads. In addition, deterministic task scheduling can better leverage advancements in the IVN –such as cross-zonal connections in the cross-zone mesh topology (see Figure 101.c)– by flexibly managing task completion deadlines to efficiently schedule tasks across the IVN.

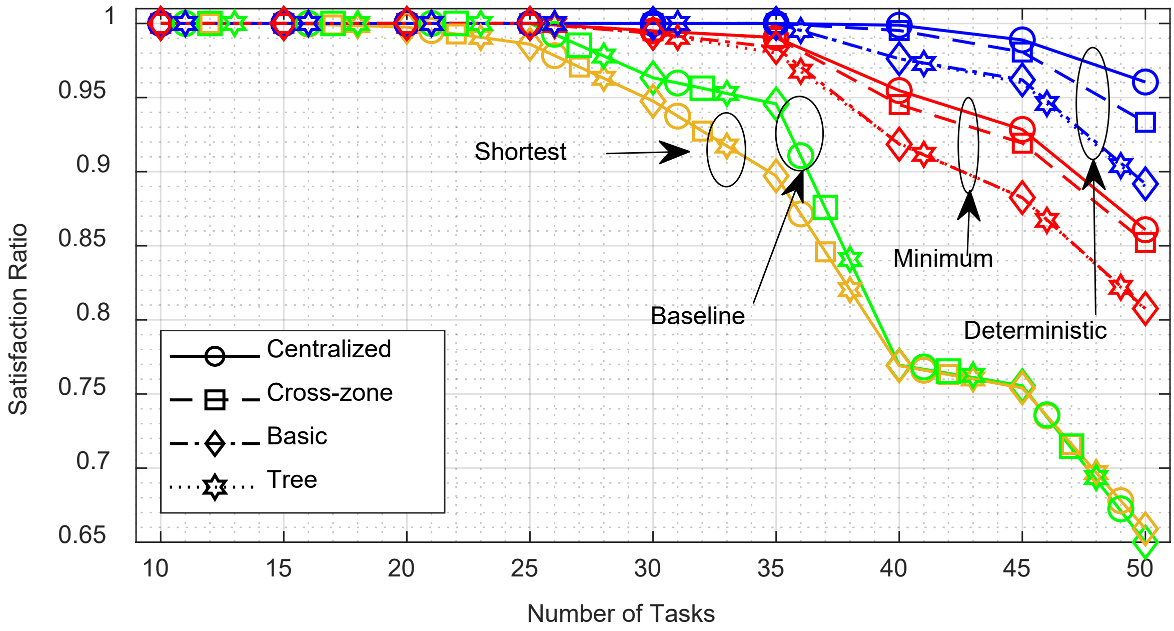


Figure 102 Task satisfaction ratio for different schemes (wired topologies).

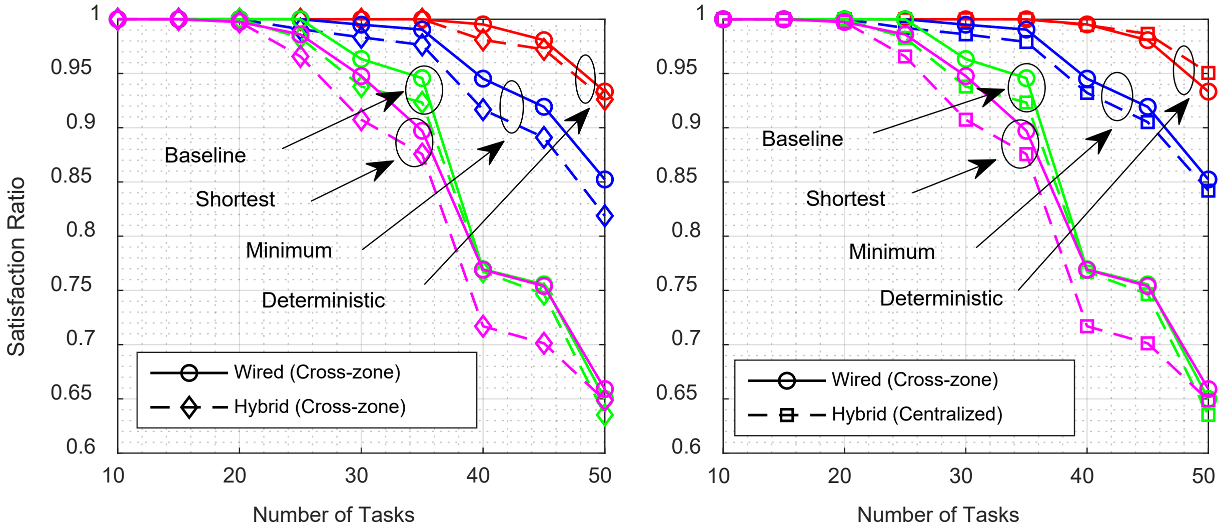


Figure 103 Task satisfaction ratio for different schemes in wired & hybrid cross-zone (left), and wired cross-zone & hybrid centralized (right).

We also analyse the impact of introducing wireless links in the cross-zone mesh IVN topology, specifically in the connections between sensors/actuators and the ECUs. Figure 103-left compares the satisfaction ratio achieved with the fully wired and hybrid wired-wireless implementations of the topology. The figure shows that all task scheduling schemes experience a reduction in the ratio of satisfied tasks with the introduction of wireless links. However, the reduction is the smallest with the Deterministic scheme. For instance, the reduction experienced with the Deterministic scheme under all considered task loads remains below 1.5%, while it increases to 4.3%, 8.6% and 3% for the Minimum, Shortest and Baseline schemes, respectively. This is because Deterministic takes the reliability of the links into account when scheduling tasks to computing units across the IVN. The introduction of wireless links improves the capacity to establish new links within the IVN, and facilitates the flexibility and reconfigurability sought with SDVs. For example, it would be possible to evolve a fully wired cross-zone mesh IVN topology to a hybrid wired-wireless implementation of the centralized mesh topology by adding a wireless link between sensors/actuators and the HPCU (Figure 101). In this case, Figure 103-right demonstrates that, with the hybrid centralized mesh IVN topology, the Deterministic scheme compensates the performance degradation resulting from the introduction of wireless connections in the hybrid cross-zone mesh IVN topology, and even achieves higher satisfaction ratios compared to the wired cross-zone mesh IVN topology. This is not actually the case for all the other schemes that fail to mitigate the impact of wireless connections, resulting in lower satisfaction ratios with the hybrid centralized mesh IVN topology than with the wired cross-zone mesh IVN topology. The results also show that the Deterministic scheme is the only scheme that achieves a satisfaction ratio above 95% in the scenario with 50 tasks in the hybrid centralized mesh IVN topology, outperforming alternative task scheduling schemes by 12.9% to 49.9%.

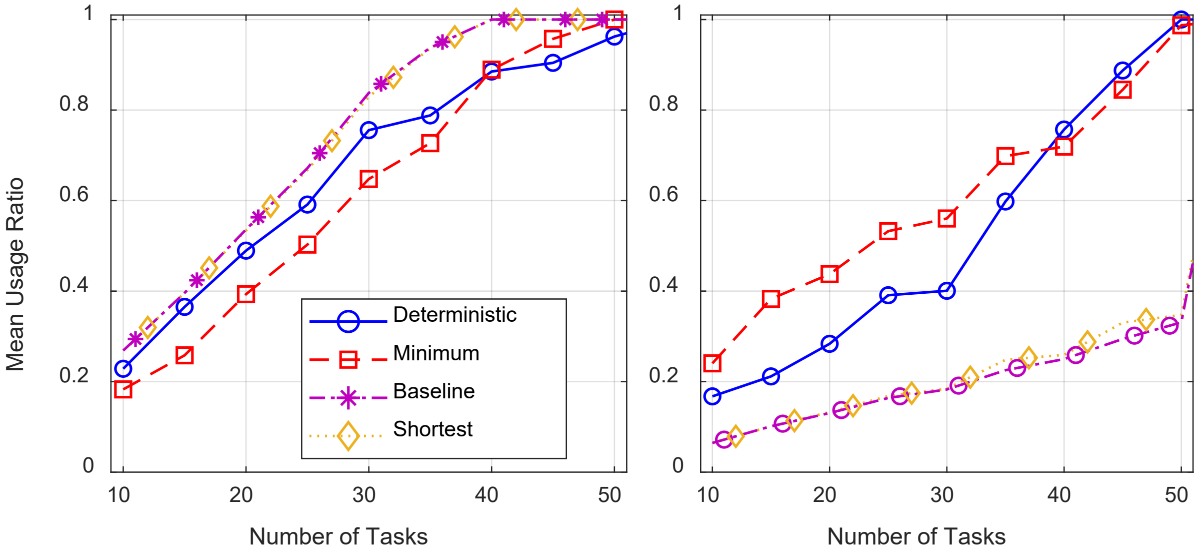


Figure 104 Usage ratio of computing units (ECUs – left, HPCU – right) in the hybrid wireless-wired centralized mesh IVN topology.

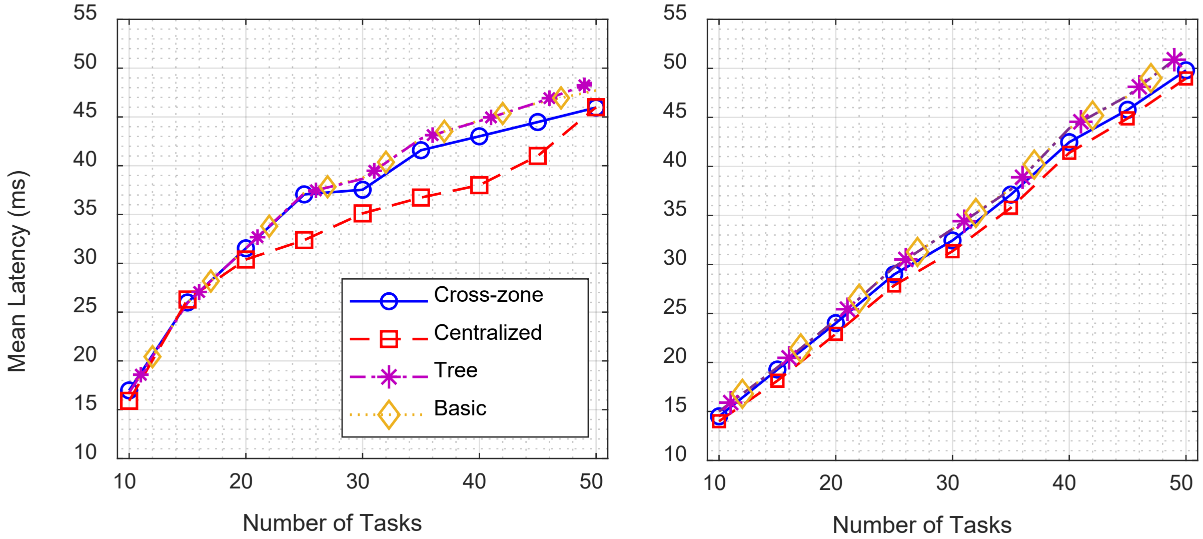


Figure 105 Latency in different topologies for the Deterministic (left) and Minimum (right) schemes.

The higher satisfaction ratios achieved with the Deterministic scheme stem from its better scheduling and more balanced workload and resource utilization across the IVN. This is illustrated in Figure 104, which depicts the average ratio of utilized computing resources of computing units in the hybrid wireless-wired implementation of the centralized mesh topology; similar trends are observed in the other topologies. The left figure shows the average usage ratio of the zone ECUs, while the right figure depicts the usage ratio of the HPCU. Figure 104 shows that the Baseline and Shortest schemes saturate the ECUs in the scenarios with 40 tasks or more, while the HPCU experiences a low usage ratio (below

25%) even when the ECUs are saturated. This saturation of the ECUs leads to the drop in the satisfaction ratio shown in Figure 103 for the Baseline and Shortest schemes. The Deterministic and Minimum schemes distribute tasks across different computing units, making better use of the HPCU's high processing power compared to the Baseline and Shortest schemes. Comparing the Deterministic and Minimum schemes, Figure 104 shows that the Minimum scheme tends to utilize more the HPCU to minimize the task execution times in scenarios with low to medium task loads. However, under higher task loads, it utilizes the ECUs more than the Deterministic scheme. In contrast, the Deterministic scheme follows the opposite trend, relying more on the HPCU at higher task loads, which helps avoid bottlenecks in the ECUs by distributing the load more efficiently. This more balanced distribution results in the higher satisfaction ratios shown in Figure 103 for the Deterministic scheme.

Finally, Figure 105 compares the latency or total execution time T_n experienced by the Deterministic and Minimum schemes under different IVN topologies. Results are reported for the wired implementation of the tree-based, basic mesh, and cross-zone mesh IVN topologies, and the hybrid wireless-wired implementation of the centralized mesh topology. Figure 105 shows that Deterministic experiences higher latency than Minimum in scenarios with low to medium task loads because it prioritizes maximizing the number of tasks completed before their deadlines (Figure 102- Figure 103) over minimizing latency. On the other hand, Deterministic reduces the latency under higher-load scenarios thanks to its capacity to efficiently adapt the tasks' scheduling to the computing workload, as shown in Figure 104. Results in Figure 103 showed that Deterministic was the task scheduling approach that could better handle the introduction of wireless links. This is also visible in Figure 105 that shows that Deterministic reduces the latency in the centralized mesh IVN topology by up to 16.3% and 15.6% compared to the tree/basic mesh and cross-zone mesh IVN topologies, while Minimum only reduces it by 8.3% and 5.5%, respectively.

4.4.7 Summary

This study has introduced a novel deterministic task scheduling scheme for in-vehicle networks and has demonstrated its potential to exploit the capabilities of in-vehicle zonal E/E architectures with centralized computing. Our analysis has demonstrated that a deterministic approach to task scheduling can better guarantee deterministic service levels than alternative approaches and can satisfactorily support be increasing in-vehicle computational workloads and tasks. This is achieved thanks to a more balanced workload distribution and resource utilization across the IVN. These trends have been validated across a variety of IVN topologies with consideration of wireless connectivity in hybrid IVN topologies.

4.5 COMPUTE AWARE TRAFFIC STEERING WITH MOBILITY CONSIDERATIONS

4.5.1 Introduction

This section addresses the problem of how the network infrastructure can steer traffic between clients of a service and sites offering the service, considering both network metrics (such as bandwidth and latency), and compute metrics (such as processing, storage capabilities, and capacity).

This might be particularly useful for use cases such as AR/VR/XR and 6G-SHINE subnetwork use cases described in [2] such as interactive indoor gaming, AR navigation and in industrial subnetworks, where not only delay is relevant, but also the computing resources required for running the service/application such as AR/VR/XR or digital twin.

4.5.2 Use Case and Scenario

Let us consider a general use case where a terminal (e.g., UE) is running an AR/VR/XR application. We consider that a part of this service is executed in the subnetwork infrastructure, posing some requirements on the connectivity (e.g., delay between the terminal and the node where the service is executed on the network infrastructure) and computing resources (e.g., capabilities to render the XR video within a certain latency budget). Within the subnetwork domain where the terminal is connected to there are multiple sites capable of hosting the service such as LC, HC or 6G BS, each with potentially different connectivity and computing characteristics. Figure 106 shows an example scenario of the 6G subnetworks where the service is running on an SNE on the left (as a terminal, marked with a red circle), and part of this service is executed on either LC or HC elements of a subnetwork.

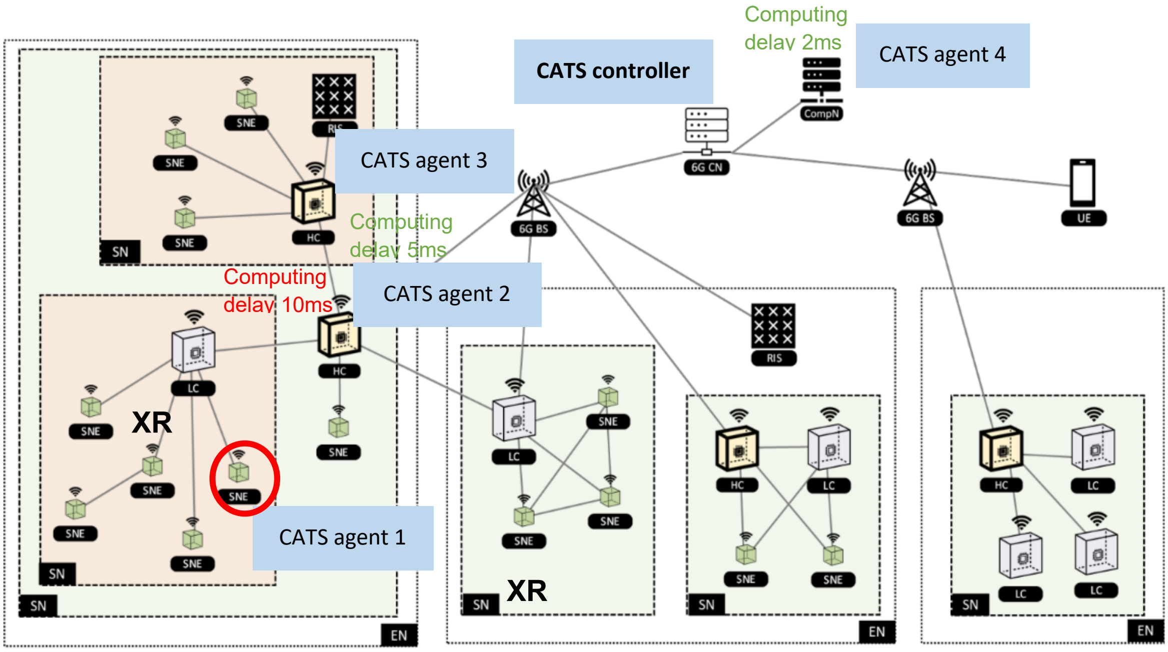


Figure 106 Compute-aware traffic steering in subnetworks

End point for the rendered XR video can be located within or outside subnetwork that generates video. In this scenario, this is done by CATS agent 1. The compute path for this device can be constituted of CATS agents 1, 2, 3 and 4, where CATS agent 1 is the constrained SNE, CATS agent 2 is located outside the SNE's subnetwork, CATS agent 3 is located within another subnetwork, and CATS agent 4 is co-located with the CN, where compute capabilities may be seen as non-depletable. Moreover, multiple subnetworks can generate video. XR application requirements are typically strict in terms of rendering (processing capabilities and availability) and latency (connectivity latency + processing latency).

4.5.3 Problem Definition

Current networking systems mainly take into consideration connectivity characteristics when deciding how to route traffic. Joint compute and networking solutions are missing. There is no network-based mechanism that enables up-to-date service instantiation decisions coupled with connectivity requirements. This problem is even more prominent in subnetworks in which different services are hosted at different subnetwork entities. Therefore, it important to consider joint computing and networking requirements for better quality of experience and quality of service for subnetwork use cases.

4.5.4 Proposed Solution

We propose solutions to enable subnetworks to select the best site to instantiate a terminal service, considering service-specific requirements at both connectivity and computing levels. We address the following questions:

- What information does the network need to be able to select the best location for a service to be instantiated?
- How to steer traffic between the terminal and the selected service site, in a way that is transparent to the network forwarding infrastructure, and even to the terminal?

To enable this, we propose CATS agent and CATS controller functionality in the subnetwork entities. These are described in the following.

4.5.4.1 CATS Agent

We propose the entities in the subnetwork to have CATS agent, and each agent has the following functionality:

- **Instance selection**: it deals with the procedures required to perform service and terminal specific instance selection. The subnetwork entities need this functionality so they can select the location of a given service instance. Optionally, a terminal (UE or SNE) might also run this engine, to actively participate in the selection process.
- **Traffic steering**: it deals with the ingress and egress entity selection and the associated traffic steering between them, to meet the connectivity and computing requirements of the service. The CATS agent functionality can also run on the terminal to aid the network deciding or actively influence its site selection.

4.5.4.2 CATS Controller

We also propose to have a CATS controller in the network residing either at the core network or closer to the subnetwork entities. The CATS controller has the overall view of all the entities (egress and ingress points) of the domain. CATS agents and CATS controller in subnetworks are shown in Figure 106 as an example.

Let us assume that LC is the ingress node that receives request from the terminal, and we assume that HC nodes in the subnetwork can have CATS functionality. However, the procedures are not only limited to such assumption. Any node can be a CATS-aware node with different CATS services hosted at each one of them. In the following we describe an extended terminal service request procedure enabling the network infrastructure to select a service instance meeting the connectivity and computing requirements of the service, and the setup of the required traffic steering for the service traffic.

A terminal requests a service (e.g., AR/VR/XR) that requires specific connectivity and computing resources (CATS requirements) as shown in Figure 107. The request is sent to an ingress node (LC in this case), including a service ID and, if the terminal is CATS-aware, a list of requirements such as latency, bandwidth, computing resources, and affinity constraints. The LC processes this request and selects an appropriate egress node (HC in this case) through one of two options:

1. **Distributed Option (Direct Query to LC node):** The LC queries all or a subset of HCs in the domain, including parameters like service ID, terminal ID, and CATS requirements. Each HC responds with its capabilities. The LC selects an appropriate HC based on these responses.
2. **Centralized Option (Query to a CATS Controller):** Instead of querying multiple ECRs, the ICR sends the request to a central CATS controller, which has a global view of all sites. The controller evaluates the best site and responds with the selected ECR's details.

Once an egress node (HC) is selected, the LC node sends a request to establish a traffic steering session, including the same information as the original CATS query. If accepted, the HC responds with an acknowledgment, confirming service details, assigned IP prefix. An IP tunnel is then established between the LC and HC nodes, with traffic forwarding configured. The LC node provides the allocated IP prefix to the terminal via Router Advertisements or DHCP, ensuring that service traffic is steered through the established tunnel.

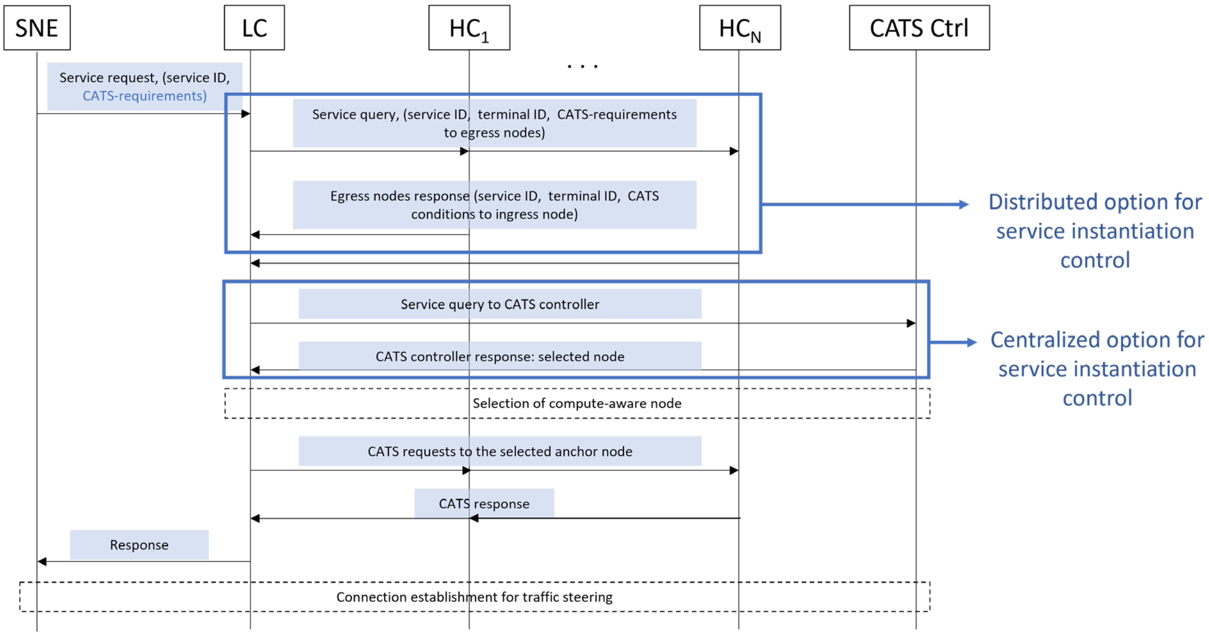

SNE
LC
HC_1
. . .
HC_N
CATS Ctrl
Service request, (service ID, CATS-requirements)
Service query, (service ID, terminal ID, CATS-requirements to egress nodes)
Egress nodes response (service ID, terminal ID, CATS conditions to ingress node)
Distributed option for service instantiation control
Service query to CATS controller
CATS controller response: selected node
Centralized option for service instantiation control
Selection of compute-aware node
CATS requests to the selected anchor node
CATS response
Response
Connection establishment for traffic steering

Figure 107 Signaling example of CATS, initiated by a CATS-aware terminal (Distributed Option)

4.5.5 Solution with Mobility Consideration

The above solution does not take mobility of the terminal into consideration before instantiating a service at a particular node or when deciding a particular node to host the service. Moreover, current mobility solutions in networking systems mainly take into consideration connectivity characteristics when taking mobility-related decisions such as service migration. Service mobility solutions jointly considering computing and networking solutions are missing.

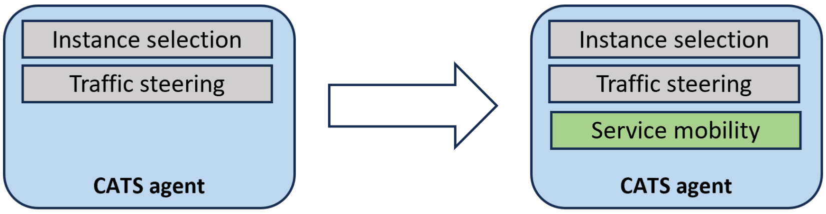


Figure 108 Extension of CATS agent functionality to mobility services

To enable this, first, CATS agents need to have functionality related to mobility. Therefore, we propose to extend the functionality of the CATS agents and CATS controller to service mobility along with instance selection and traffic steering functionality as shown in Figure 108

- **Service mobility:** it deals with the procedures required to (*i*) detect or predict a change of the current conditions, jointly considering computing and networking, requiring of a service mobility operation; (*ii*) selecting the best target service instance location, and (*iii*) triggering the service mobility by orchestrating the service anchor mobility and requesting service migration to a new site. For example, a terminal or HC node (egress node) might use this functionality to perform active monitoring of a service with CATS agents running at the current LC, HC nodes and or service site. It is also used to perform the actual service anchor mobility.

The service mobility can be triggered by CATS-aware terminal as shown in Figure 109, CATS-agent or CATS controller. In the same way as we have provided solution for traffic steering and service instantiation, by having a CATS agent running on the terminal (SNE) or at LC or HC nodes, it can perform different monitoring actions to predict or detect the need to migrate a service from one site to another. This CATS agent might, for example, interact with other CATS agents deployed on other subnetwork entities.

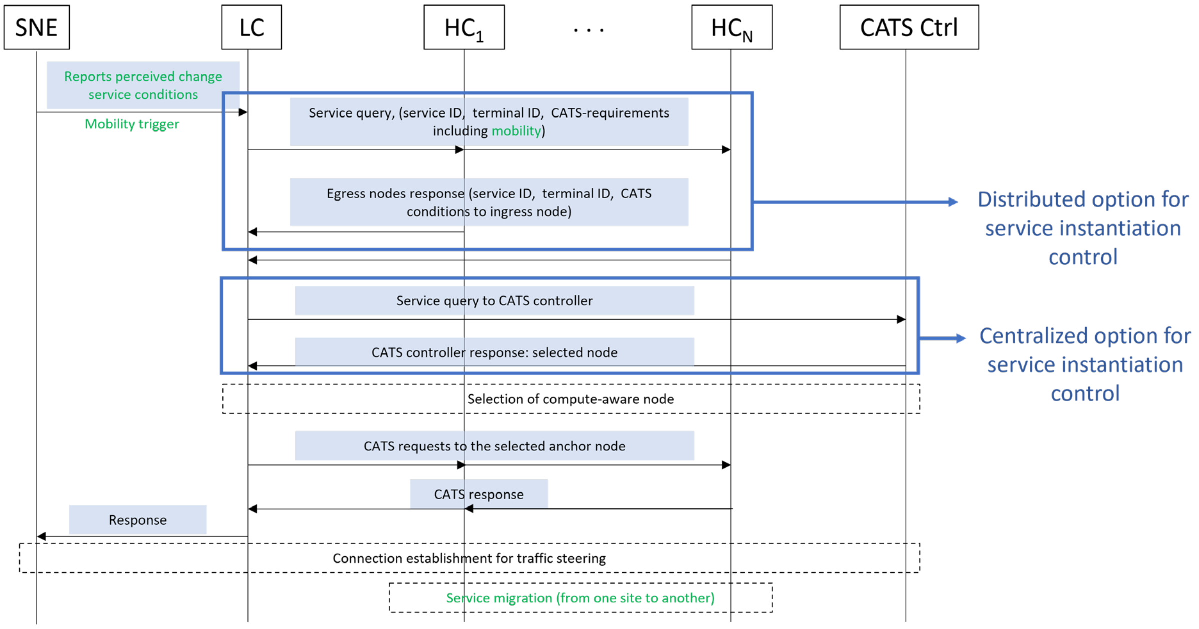

SNE
LC
HC_1
. . .
HC_N
CATS Ctrl
Reports perceived change service conditions
Mobility trigger
Service query, (service ID, terminal ID, CATS-requirements including mobility)
Egress nodes response (service ID, terminal ID, CATS conditions to ingress node)
Distributed option for service instantiation control
Service query to CATS controller
CATS controller response: selected node
Centralized option for service instantiation control
Selection of compute-aware node
CATS requests to the selected anchor node
CATS response
Response
Connection establishment for traffic steering
Service migration (from one site to another)

Figure 109 Example Signaling, initiated by CATS-aware terminal with mobility considerations (Distributed Option)

In a service anchor mobility procedure for CATS, initiated by a CATS-aware terminal. the network infrastructure is capable to select a target service instance meeting the connectivity and computing requirements of the service, with signalling procedures defined to perform a transparent anchor migration to a new site, facilitating the service migration in a transparent way for the terminal.

4.5.6 Summary

In this work we proposed the CATS framework, enabling joint compute and network-aware service selection for latency-sensitive applications like AR/VR/XR in 6G subnetworks. The CATS framework enables joint compute and network-aware traffic steering through the introduction of two key functional entities: CATS agents and CATS controllers. The proposed system enhances service instance selection and traffic steering by dynamically selecting the best service site (e.g., LC or HC nodes) based on real-time connectivity and computing constraints. This improves Quality of Experience (QoE) and Quality of Service (QoS) by ensuring optimal service execution in heterogeneous, resource-constrained, and mobility-prone environments. Furthermore, the extension of CATS to support mobility-aware service anchoring and migration ensures seamless service continuity for mobile terminals in subnetworks.

5 DYNAMIC SPECTRUM SHARING

The evolution towards 6G networks introduces new architectural and operational challenges, particularly in the context of emerging "in-X" subnetworks such as in-factory industrial networks, in-vehicle communication systems, and dense IoT clusters. These subnetworks demand highly reliable and low-latency connectivity, necessitating flexible spectrum access and novel device communication paradigms. In Section 5.1, two key enablers for such subnetworks are addressed: spectrum regulation and device autonomy. The first part analyses global approaches to spectrum sharing and policy, highlighting regulatory innovations and emerging bands relevant to 6G. Currently spectrum sharing happens in 3GPP SL, where the BS assigns the SL resources. Section 5.2 critically evaluates current 3GPP SL mechanisms for UE-to-UE communication, identifying limitations in network-controlled SL Mode 1 and the potential of more autonomous SL Mode 2 operations. To address these limitations a method for dynamically assigning network-controlled licensed resources for SN use is then presented in Section 5.2. Together, these perspectives provide insights into how future 6G subnetworks can achieve scalable, efficient, and context-adaptive connectivity.

5.1 DYNAMIC SPECTRUM SHARING AND REGULATION

5.1.1 Introduction

6G in-X subnetworks might potentially demand flexible spectrum access [2]. Unlike traditional cellular macro-networks, these specialized subnetworks (e.g. in-factory 6G networks for Industry 4.0, in-vehicle networks for autonomous cars, or dense IoT clusters) require ultra-reliable, low-latency links tailored to their environment [2]. Achieving this performance necessitates innovative spectrum-sharing mechanisms and supportive regulatory frameworks. This section provides a comprehensive analysis of how the European Union (EU), China, and the United States (US) might approach spectrum regulation and sharing for such in-X subnetworks. We compare licensed and license-exempt spectrum policies, evaluate sharing techniques (from EU's Licensed Shared Access to the US CBRS three-tier model), review compliance and enforcement, and discuss emerging trends. Finally, we outline potential new 6G frequency bands (e.g. mid-band expansions and sub-THz ranges) and their implications for future spectrum policy.

5.1.2 Regulatory Frameworks

Each region has developed distinct regulatory frameworks balancing exclusive licensed allocations with shared or license-exempt access. Table 1 summarizes key spectrum policies in the EU, China, and US, focusing on provisions relevant to industrial, automotive, and IoT 6G subnetworks.

European Union: The EU follows a harmonized approach via CEPT/ECC decisions and national regulators, blending exclusive licensing for mobile operators with new sharing models and unlicensed bands. Traditionally, mobile spectrum (e.g. 3.5 GHz for 5G) is auctioned to carriers, but EU regulators have introduced Licensed Shared Access (LSA) frameworks and local licensing to support vertical industries. LSA, standardized by ETSI, is a two-tier sharing model where a secondary licensee (e.g. a mobile operator or enterprise) can access spectrum when the primary incumbent (e.g. military or satellite user) is not using it [27]. The LSA system uses a central database (LSA repository) with mostly static incumbent information [27]. Early trials in Europe targeted the 2.3–2.4 GHz band for LSA [27], though adoption has been slow. In parallel, several EU countries opened spectrum for private 5G/6G networks: for example, Germany reserves 100 MHz in the 3.7–3.8 GHz band for local industrial networks [28], granting site-specific licenses (in 10 MHz blocks, up to the full 100 MHz) to companies for up to 10 years [28]. France has allocated a 40 MHz slice in 2.6 GHz for industrial IoT and is exploring local licensing in 3.8–4.2 GHz [29]. UK established a Shared Access License regime covering 1800 MHz, 2.3 GHz, 3.8–4.2 GHz, and even 24.25–26.5 GHz (indoor) for localized use [30]. These EU initiatives enable factories, ports, and campuses to deploy their own 5G/6G networks outside the mobile operators' exclusive spectrum [28]. On the license-exempt side, Europe designates bands like 863–870 MHz (for IoT short-range devices), 2.4 GHz and 5 GHz (Wi-Fi/ISM), and recently 5925–6425 MHz for Wi-Fi 6E. Use of such bands requires compliance with technical limits (e.g. ≤25 mW and duty cycle limits in 868 MHz, dynamic frequency selection (DFS) in 5 GHz to avoid radars) but no individual license [30]. The EU (CEPT) opted not to open the upper 6 GHz (6425–7125 MHz) for unlicensed use, instead favoring IMT (mobile service) allocation – a direction confirmed by WRC-23 where 6425–7125 MHz is being identified for licensed 5G/6G in Europe [31]. For automotive applications, the EU mandates a dedicated ITS band at 5.9 GHz (5855–5925 MHz) for vehicle-to-vehicle and vehicle-to-roadside communications; this is a licensed-exempt band but restricted to intelligent transport systems, coordinated by standards (ETSI ITS-G5 or C-V2X) rather than a dynamic sharing database.

United States: The US has been a pioneer in dynamic spectrum sharing through the FCC's Citizens Broadband Radio Service (CBRS) and other frameworks. In CBRS (3.55–3.7 GHz band), a three-tiered access model is managed by a Spectrum Access System (SAS) database [30]. Incumbents (primarily Navy radars and fixed satellite stations) form Tier 1 with highest priority and full protection. Tier 2 consists of Priority Access License (PAL) holders (e.g. companies that won county-level licenses at auction, up to 70 MHz total), who receive interference protection from lower-tier users [32]. Tier 3 is General Authorized Access (GAA), open to any certified user/equipment, which can use the spectrum opportunistically with no interference protection (must accept interference from higher tiers) [30]. The SAS dynamically assigns frequencies to PAL and GAA users in real-time based on availability, ensuring incumbents are not affected. This CBRS scheme has unlocked 150 MHz of mid-band spectrum for private LTE/5G networks (industrial IoT, rural broadband, etc.) that previously was underutilized by federal incumbents. It is cited as a breakthrough in spectrum sharing, marrying cooperative sharing (through centralized control) with market mechanisms [27]. Beyond CBRS, the US also allows TV White Space devices in vacant TV channels (using geolocation databases to avoid broadcast incumbents), and in 2020 the FCC opened the entire 6 GHz band (5925–7125 MHz) for unlicensed use. Standard-power 6 GHz Wi-Fi in the US must engage an Automated Frequency Coordination (AFC) system (a database) to avoid interfering with licensed point-to-point microwave links, another example of database-assisted sharing. For licensed mobile spectrum, the US generally awards exclusive-use licenses via auctions (e.g. 3.7–3.98 GHz C-band for 5G). Unlike the EU, the US has not broadly issued local licenses to enterprises in mid-band – instead, it relies on frameworks like CBRS GAA for enterprises to access spectrum. Automotive communications in the US are regulated in the 5.9 GHz ITS band; recent FCC rulings transitioned 30 MHz of this band to cellular V2X technology and repurposed the other 45 MHz for Wi-Fi, effectively sharing the band between vehicular comms and unlicensed use (though on separate sub-bands). License-exempt IoT use is permitted in ISM bands (902–928 MHz, 2.4 GHz, 5 GHz, etc.) under FCC Part 15 rules, with strict emission limits and a non-interference condition (unlicensed users must accept any interference from licensed services) [30].

China: China's spectrum regulation remains centered on exclusive allocations to state-owned mobile operators, with relatively limited provisions for independent private networks. All cellular spectrum (e.g. 2.6 GHz, 3.5 GHz, 4.9 GHz for 5G) is licensed to the major carriers (China Mobile, China Telecom, China

Unicom, and China Broadcasting Network). Unlike Europe or the US, Chinese regulators (MIIT) currently do not issue spectrum directly to enterprises or permit CBRS-style open access [33]. Private 5G for industries is delivered via the operators – often through network slicing or dedicated infrastructure provided by the carrier on their licensed frequencies [33]. For instance, by end of 2020, Chinese operators reported 800+ private 5G network deployments for factories, mines, ports, etc., but these all leverage operator-held spectrum. This “three-cornered game” means enterprises must “make do with what the operators offer”. However, China is experimenting with intra-operator sharing and co-investment models. Notably, MIIT assigned the 3300–3400 MHz band to three operators on a shared basis – China Telecom, China Unicom, and China Broadcasting Network each have rights, with the band meant for indoor 5G use only [30]. This is the first instance of Chinese regulators explicitly allowing shared use among multiple licensees in the same band. The indoor-only restriction and coordination agreements aim to mitigate interference while maximizing utilization of a band that would otherwise lie fallow or be unevenly used by a single operator. Beyond licensed bands, China does allow license-exempt usage in certain ranges for IoT and wireless LANs – e.g. 2.4 GHz and 5.8 GHz are open for Wi-Fi (with equipment type-approval), and China has specific IoT bands like 779–787 MHz and 430–433 MHz for short-range devices. A unique aspect is the sub-GHz IoT band 470–510 MHz, widely used in China for LPWAN technologies (e.g. LoRa), which is effectively a quasi-ISM band allocated for telemetry and IoT. These unlicensed or lightly licensed bands in China are governed by technical regulations (power limits, duty cycle, etc.) similar to other countries’ ISM rules. For automotive V2X, China has allocated 5905–5925 MHz for C-V2X direct communications, aligning with its push for connected autonomous vehicles, and this band is managed by MIIT with usage rights typically granted to automotive OEMs and operators in a controlled manner.

Table 2 Spectrum Policy Comparison (EU, China, US)

Aspect / Region	European Union	China	United States (US)
Mobile Spectrum Licensing	Primarily exclusive national licenses for MgtNOs (e.g. 3.4–3.8 GHz for 5G). Some bands (e.g. 700MHz, 3.5 GHz) allocated via auctions to operators under EU-wide frameworks.	All mobile spectrum allocated to state-owned carriers (no independent MgtNOs). Exclusive national assignments (e.g. 2.6, 4.9 GHz for 5G) to China Mobile/Telecom/Unicom; China Broadcasting Network added for 5G broadcast.	Exclusive licenses auctioned to nationwide operators (Verizon, AT&T, etc.) for key bands (600 MHz, C-band 3.7–3.98 GHz, 28 GHz, etc.). Strict build-out and use requirements to prevent warehousing.

Dedicated Spectrum for Vertical/Private Networks	Several countries issue local 5G licenses for industries. *E.g.* Germany 3.7–3.8 GHz (100 MHz) for campus networks; UK Shared Access licenses (1.8, 2.3, 3.8–4.2 GHz) on FCFS basis; France 2.6 GHz for industrial IoT. EU-level support for vertical spectrum, but implementation is national.	No direct licensing to enterprises (private entities cannot obtain spectrum). Vertical 5G is delivered via network slicing or operator-run private networks. *Exception:* 3300–3400 MHz allocated jointly to 3 operators for indoor use , effectively a shared operator band to support industrial deployments.	CBRS (3550–3700 MHz) opens 150 MHz for private and rural broadband via tiered sharing (enterprises can use GAA tier freely). No dedicated *licensed* spectrum set aside for private 5G nationwide, but localized use of unused licensed spectrum possible via leasing or FCC's experimental licenses. Some states and utilities use 900 MHz narrowband for private LTE under recent FCC realignment.
Licensed Shared Access (LSA) & Dynamic Sharing	**LSA:** Two-tier sharing framework (incumbent + licensed secondary) standardized by ETSI. Trialed at 2.3 GHz; concept extended to others. Not widely commercialized yet (regulators exploring evolved LSA with more dynamic features). Spectrum leasing/trading: allowed in EU policy – operators can sub-lease spectrum to third parties (though uptake has been limited).	No formal LSA or CBRS-like frameworks for dynamic sharing with databases. Spectrum sharing occurs via operator coordination (e.g. co-build agreements). Some research on dynamic spectrum (China's IMT-2020/2030 promotion group studies AI-driven spectrum management) but regulatory action lags.	**Three-tier CBRS SAS** with dynamic frequency assignment. TV White Space database system for unused TV channels (since 2010). FCC 6 GHz AFC for protecting incumbents while allowing unlicensed. Dynamic Spectrum Sharing (DSS) technology (not regulatory, but MgtNO-driven) allows 4G/5G coexistence in same band. New proposals to share federal bands (e.g. 3.1–3.45 GHz) with commercial use via automated coordination

Unlicensed / License-Exempt Spectrum	**ISM bands:** 868 MHz (Europe-specific IoT band), 2.4 GHz, 5 GHz (5470–5725 MHz with DFS). Wi-Fi 6E: 5925–6425 MHz opened for RLAN (lower half of 6 GHz). 60 GHz mmWave: 57–71 GHz available for unlicensed use (very low range, used for 5G NR-U, WiGig). EU devices must meet harmonized standards (ETSI EN 300 328, etc.) limiting power and requiring LBT/DFS to prevent harmful interference	**ISM bands:** 2400 MHz and 5100–5875 MHz for WLAN; sub-GHz bands like 433 MHz and 779 MHz for short-range IoT. LPWAN: 470–510 MHz allocated for unlicensed IoT (LoRaWAN, etc.) within technical constraints. 60 GHz: opened for unlicensed (used for 5G local high-throughput links). Chinese regulations for unlicensed are strict; equipment must obtain MIIT approval and users must accept interference from licensed services.	**ISM bands:** 915 MHz, 2.4 GHz, 5 GHz widely used for IoT/Wi-Fi. Wi-Fi 6E: entire 5925–7125 MHz opened for unlicensed (indoor low-power across band; standard-power with AFC to protect incumbents). 57–71 GHz unlicensed (WiGig/802.11ad). Devices are subject to FCC Part 15 rules (power/EIRP limits, e.g. 4 W EIRP in 5.8 GHz ISM, much lower in sub-GHz) and must not cause interference to any licensed service.
Automotive V2X & ITS Spectrum	5855–5925 MHz reserved for ITS (safety messages between vehicles and infrastructure). EU initially used ITS-G5 (802.11p), now also allowing C-V2X; coordination through profile standards rather than dynamic allocation. Not a general-purpose subnetwork band, but important for automotive IoT.	5905–5925 MHz allocated for C-V2X direct communication (LTE-V2X) for connected vehicles. China aggressively mandates C-V2X in new vehicles; spectrum rights managed via government and auto industry partnerships. Sharing not applicable – band exclusively for ITS nationwide.	5850–5925 MHz was Dedicated Short Range Communications (DSRC). In 2020, FCC repurposed 5925–5875 MHz for Wi-Fi, keeping 5905–5925 MHz for C-V2X. The 30 MHz for V2X is licensed by rule (no individual licenses; devices authorized under Part 95). Now essentially a carve-out band for automotive safety, with remaining part shared with unlicensed Wi-Fi.

In summary, the EU fosters a mixed regime – continuing to allocate exclusive spectrum to operators for wide-area 5G/6G, but also promoting sharing via LSA and local licenses to open spectrum access to industrial players. The US leans on market-driven sharing frameworks like CBRS, harnessing databases and tiered access to accommodate both incumbent protection and new entrants. China so far emphasizes centralized control and operator-led deployments, with limited direct sharing for private enterprises.

5.1.3 Spectrum Sharing techniques

Spectrum sharing techniques can be broadly categorized by their coordination approach and regulatory underpinnings. Key mechanisms include Licensed Shared Access, CBRS-like tiered frameworks, cooperative vs. non-cooperative sharing, and advanced dynamic allocation methods. Below, we evaluate these mechanisms and their relevance in EU, China, and US for industrial, automotive, and IoT subnetworks.

5.1.3.1 Licensed Shared Access

Licensed Shared Access is a coordinated sharing method where an incumbent license-holder (e.g. government or legacy user) permits a licensed entrant to use its band under defined conditions. It's essentially a two-tier system – incumbent (Tier 1) and LSA licensee (Tier 2) – enforced by agreements and a spectrum coordination system [27]. EU Perspective: LSA was championed in Europe to unlock underused bands like 2300–2400 MHz (held by militaries in some countries) for cellular use without fully clearing the incumbents [27]. ETSI specified LSA system architecture with an LSA Controller (manages spectrum on the operator side) and an LSA Repository (database of incumbent usage). The LSA Repository stores where/when the primary user operates; the Controller grants frequencies to the secondary user elsewhere or at other times. Early implementations treated incumbent usage as relatively static (e.g. an incumbent using certain areas persistently) [27]. While LSA ensures predictable, licensed-quality access for the secondary (unlike unlicensed, the LSA licensee gets exclusive use when granted), it requires tight cooperation between stakeholders and regulatory backing. Europe's trials showed technical feasibility, but full deployment stalled due to complexities in agreements and lack of incumbent incentives [34]. Now, regulators are revisiting LSA for new bands (e.g. considering evolved LSA (eLSA) with more real-time sensing to detect incumbent activity) [27]. US/China: The US did not implement LSA per se, opting for the more granular CBRS model. China hasn't publicly adopted LSA, though conceptually the 3300–3400 MHz shared assignment among operators is a form of licensed co-sharing. In 6G, LSA could be expanded to additional bands – for example, sharing military mmWave bands or satellite spectrum with industrial 6G networks on a localized basis, using automated coordination. LSA's strength is providing quality guarantees (since the sharing is managed and licensed), which is vital for industrial subnetworks that can't tolerate random interference. Its weakness is the administrative overhead and potential inflexibility if incumbents have unpredictable usage (something eLSA aims to solve via dynamic sensing).

5.1.3.2 CBRS and Multi-Tier Dynamic Sharing

The CBRS model demonstrates a three-tier, dynamic sharing regime with automated enforcement [30]. A cloud-based Spectrum Access System (SAS) continuously assigns frequencies to users based on priority and real-time incumbent availability. This is a cooperative sharing approach: all devices register with the SAS and cooperate by following its spectrum grants. The SAS also interfaces with Environmental Sensing Capability (ESC) sensors along coastlines to detect Navy radar operations, automatically vacating frequencies for incumbents when a radar is active. Pros: Highly efficient utilization – spectrum is idle for minimal time – and granular sharing (down to small geographic areas or short time slots if needed). Cons: System complexity and reliance on network connectivity (devices must contact SAS). US Perspective: CBRS is fully operational in the US, supporting private LTE/5G in enterprises, IoT networks, and WISP (wireless ISPs) services. The framework has been held up as a model for other bands and countries [27]. For 6G, the CBRS concept could extend to new frequency ranges and even new tier structures (e.g. two-tier in some bands, three-tier in others). EU Perspective: Europe has not implemented a CBRS clone yet, but there is interest in dynamic database-driven sharing for, say, 6 GHz standard-power Wi-Fi (the AFC system, which is conceptually similar to SAS). Additionally, EU research projects consider multi-tenant 6G networks where a neutral host could dynamically allocate spectrum to different subnetworks (akin to tiered sharing among industrial players on a campus). China Perspective: China has been cautious about multi-tenant spectrum sharing; however, as 6G pushes into high frequencies, Chinese researchers are investigating AI-driven spectrum orchestration where networks dynamically negotiate spectrum use. In practice, any CBRS-like system in China would likely be government-run (e.g. a national database) with operators feeding in usage data. For example, a cooperative sharing scenario in a smart city could involve municipal IoT networks and mobile operators sharing a 6G band via a centralized coordinator – a concept aligning with China's centralized ethos.

5.1.3.3 Cooperative vs. Non-Cooperative Sharing

A fundamental distinction in spectrum sharing is whether users actively coordinate (cooperate) or act autonomously (non-cooperative) while obeying general rules. Cooperative sharing involves explicit exchange of information or control by a central entity – e.g. LSA's incumbent informs the licensee of availability, CBRS devices talk to SAS, or operators mutually plan frequencies in a shared band. Non-cooperative sharing relies on protocols or etiquette that allow independent users to coexist without direct communication. The classic example is license-exempt Wi-Fi, where devices use listen-before-talk (LBT) and random backoff to avoid collisions in the 2.4/5 GHz band. Here, each device follows rules (like "don't transmit if someone else is transmitting") but there is no central controller – this is decentralized spectrum sharing. Another non-cooperative mechanism is dynamic frequency selection (DFS) mandated in 5 GHz: Wi-Fi APs must listen for radar signals and vacate the channel if a radar is detected. They do this autonomously, effectively "respecting" the incumbent without any direct coordination – the onus is on the device's sensing capability. In industrial 6G subnetworks, non-cooperative sharing might mean, for example, a factory 6G router using an unlicensed band, relying on spectrum sensing to avoid interfering with a nearby private network in the same band. EU vs. US vs. China: All three regions employ non-cooperative sharing in their unlicensed bands (via technical rules). The EU explicitly requires LBT in certain bands (a form of decentralized cooperation among equals). The US Part 15 rules essentially enforce a non-cooperative regime – devices must not exceed power limits and must accept interference. China's unlicensed usage similarly depends on devices adhering to power limits and channel protocols (often aligning with IEEE 802.11 or LoRaWAN specifications). Cooperative sharing, on the other hand, is exemplified by LSA in EU and SAS in US. China's only current cooperative sharing is within the operator realm (the coordination in 3300–3400 MHz among operators). For 6G, a likely trend is hybrid approaches: for instance, devices might use fast sensing and AI (non-cooperative, decentralized) but also register with a database for certain protections (cooperative). Cooperative methods excel in managing interference proactively (great for QoS in critical IoT), while non-cooperative methods excel in scalability and simplicity (no central point needed, works for massive numbers of simple IoT devices). 6G may combine these: e.g. a 6G IoT device might first check a database for a clear channel, then also perform local sensing to ensure no hidden incumbents – effectively layering cooperation and autonomy.

5.1.3.4 Dynamic Spectrum Allocation & Emerging Techniques

Dynamic allocation refers to real-time or on-the-fly assignment of frequencies to users or services based on demand, network conditions, or policies. Traditional spectrum assignments were static (fixed bands per operator). 5G introduced some flexibility (e.g. dynamic spectrum sharing technology to run LTE and 5G in one band, or carrier aggregation across licensed/unlicensed). 6G is expected to push dynamic allocation further, possibly with cognitive radio techniques and even automated negotiation between networks. Mechanisms under exploration include: (a) AI-driven spectrum brokers that monitor spectrum use in an area and dynamically grant micro-slices of spectrum to different subnetworks (with milliseconds decisions); (b) blockchain-based sharing for trust and automation in spectrum leasing (being researched to allow dynamic, short-term spectrum leases encoded in smart contracts); (c) ultra-flexible radios capable of hopping across a wide range of frequencies on demand, enabling devices to switch bands as they become available. Regulators in all regions are studying how to enable such agility while maintaining order. For instance, the US NTIA (which manages federal spectrum) has demonstrated advanced spectrum-sharing prototypes to let 5G systems opportunistically use DoD radar bands in real time [35]. China's IMT-2030 (6G) group similarly identifies dynamic spectrum management as a key enabler for dense IoT. One concrete approach is dynamic spectrum partitioning: imagine a band that is ordinarily used by a mobile operator across a city, but in the vicinity of a smart factory, that band is dynamically split so the factory's 6G subnetwork gets a chunk while the operator's macro network uses the remainder, with the split adjusted in real-time based on interference measurements. Achieving this requires both technical standardization (radio interfaces that can quickly reconfigure) and regulatory flexibility (frameworks to allow such time/space-flexible assignments).

5.1.3.5 Licensed vs. Unlicensed for In-X Subnetworks

A critical consideration is whether industrial/automotive/IoT subnetworks should operate in licensed spectrum (for reliability and control) or license-exempt spectrum (for cost and ease of deployment). The trend is towards spectrum sharing models that combine the reliability of licensing with the flexibility of unlicensed access. For example, private 6G networks in factories might use a local licensed slice of spectrum (ensuring interference-free operation within the site), obtained via a sharing framework (LSA, CBRS, or a local license from the regulator). Simultaneously, less critical IoT sensors could use unlicensed bands with adaptive protocols. Automotive subnetworks (e.g. an in-car network linking the car's sensors, passengers' devices, and road infrastructure) will likely leverage a mix: safety communications in a protected ITS band, high-bandwidth passenger services on unlicensed mmWave, and vehicular cloud connectivity via the cellular network. Global best practice appears to be converging on the notion that no single approach fits all needs – hence a toolbox of sharing techniques is required.

5.1.4 Policy and Compliance Considerations

Spectrum sharing in 6G brings not only technical challenges but also regulatory policy and compliance questions. This section identifies key regulatory constraints, enforcement mechanisms, and emerging policy trends affecting in-X subnetworks across the EU, China, and US.

5.1.4.1 Interference Protection & Power Limits

All regulators enforce limits to prevent harmful interference. In licensed sharing (LSA, CBRS), the constraint is typically do-not-interfere-with-higher-tier – secondary users must either vacate or lower power when an incumbent is present. In CBRS, this is enforced by SAS grants that simply do not authorize frequencies where interference would occur. In LSA, the license terms specify exclusion zones or times to protect the primary. For unlicensed devices, rules like power spectral density limits, duty cycle limits (in some IoT bands), and sensing requirements (DFS) serve to minimize interference risk. For example, a license-exempt 6G IoT sensor network in the EU might be limited to 25 mW EIRP in the 868 MHz band and a maximum 1% duty cycle, ensuring it cannot overwhelm other users in that band. Automotive ITS devices have their own constraints: C-V2X units in the 5.9 GHz band must adhere to power and mask limits set by regulators (22 dBm EIRP in EU, 23 dBm in US for vehicle transmitters typically) to keep interference within acceptable bounds for that safety-critical channel [30].

5.1.4.2 Enforcement Mechanisms

Enforcement in spectrum sharing can be proactive (through technological means) or reactive (through regulatory action post-violation). Proactive enforcement is exemplified by automated systems: e.g. the SAS will not grant a channel to a CBRS device if it would cause interference, effectively enforcing rules in real time. Similarly, an AFC system in 6 GHz will only provide frequencies to a Wi-Fi AP that are clear of licensed links. Reactive enforcement involves regulators monitoring spectrum and penalizing violations. Agencies like FCC and national regulators in Europe have monitoring systems (field sensors, complaint-driven investigations). If an industrial subnetwork were to operate outside its authorized parameters (e.g. a factory's private 6G network causing interference beyond its campus), the regulator could issue fines, revoke the license, or confiscate equipment. In practice, the complexity of 6G sharing might necessitate new automated monitoring – possibly even requiring that devices report their spectrum usage to a regulator portal. One interesting compliance tool is certification: devices must be certified (FCC certification in US, CE marking in EU) to ensure they implement necessary sharing protocols (like DFS, LBT). A non-compliant device (say a rogue transmitter that ignores the SAS or fails to listen in unlicensed band) is essentially illegal to operate. For industrial IoT, compliance can also be contractual – companies obtaining local spectrum licenses commit to certain usage constraints (as Germany required efficiency plans for 3.7 GHz licensees) [28]. Some regulators use automated revocation: in CBRS, if an incumbent radar is detected, the SAS can immediately tell GAA devices to cease transmissions on that frequency – this is an automated enforcement of the Navy's priority rights.

5.1.4.3 Key Regulatory Constraints

One constraint is incumbent rights, sharing frameworks must guarantee incumbents (like military radars, satellites, government users) can operate without degradation. This often limits where sharing is allowed (e.g. only indoor use or low-power use in certain bands, as China did for 3300–3400 MHz sharing). Another constraint is international coordination: spectrum policy is globally harmonized in many bands via the ITU. For instance, if Europe identifies a band for 6G IMT and China does too, but the US keeps it unlicensed, devices and standards have to accommodate divergent rules. Regulators are constrained by WRC (World Radiocommunication Conference) decisions which set broad allocations (though not legally forcing national decisions, WRC outcomes strongly influence national policies). For automotive, a policy constraint is safety – regulators are wary of allowing any other use of the ITS band that could disrupt life-saving communications, hence they hesitate to share that band with non-safety applications (the US's partial reallocation of 5.9 GHz to Wi-Fi was controversial on these grounds). Security is another emerging regulatory focus: with many IoT subnetworks potentially sharing spectrum, regulators may impose constraints for national security (e.g. requiring database systems to be secure, or prohibiting certain users/equipment vendors in shared bands).

5.1.4.4 Emerging Policy Trends

A notable trend is "use it or share it" policies [27]. Regulators are considering rules that if a license holder is not using spectrum in a location or time, it should be made available to others. The UK's Local Access License embodies this by letting others apply to use an MNO's spectrum in an area the MNO isn't serving. The FCC is similarly exploring such mechanisms for bands like 3.7–4.2 GHz (C-band) post-satellite clearance – unused parts could potentially be opened for temporary use. Automation & AI in regulation is another trend: regulatory bodies are investing in systems that can dynamically coordinate spectrum (essentially taking SAS/AFC to the next level) using AI to predict interference and manage spectrum allocation with minimal human intervention. This could lead to real-time spectrum exchanges or markets controlled by algorithms under regulator oversight. Greater sharing in millimeter-wave and sub-terahertz bands is also on the agenda, as discussed later , policymakers see these bands as an opportunity to start fresh with sharing-first frameworks (since propagation is short, frequency reuse can be high, enabling sharing by design). Lastly, there's a trend of convergence in standards: Wi-Fi and 5G/6G are increasingly overlapping (e.g. 5G NR-U using unlicensed bands), forcing regulators to consider holistic policies that cover multiple technologies sharing the same spectrum. Future 6G policies may be less about "this band is for this service" and more about technology-neutral allocations with sharing requirements that any technology must follow (e.g. a band could be open to either cellular or Wi-Fi or industrial IoT, as long as all follow a common etiquette or coordination scheme).

5.1.4.5 Compliance in Industrial and Automotive Contexts

We expect that Industrial 6G networks will often operate in controlled environments, which can simplify compliance (a factory can ensure all its devices meet the spectrum rules). Regulators may push compliance responsibility to the enterprise in such cases – e.g. a factory with a local license must police its own devices to ensure they don't interfere outside the premises, possibly through periodic audits. Automotive networks pose a different challenge: vehicles move across jurisdictions and their radios must comply on the fly. Thus, automotive spectrum compliance is largely handled in the device type approval phase (ensuring the onboard unit follows power limits, masks, etc. everywhere). A current policy debate is whether to allow cellular network operators to use automotive spectrum for general use when not needed for safety – most regulators say no, to keep it exclusive for low-latency safety messages (China and EU maintain exclusive ITS band use; US now splits the band rather than time-share it). This highlights how policy can prioritize certain applications (safety, in this case) with an absolute priority, not subject to dynamic sharing with others.

5.1.5 Global Best Practices and Impact on Spectrum Sharing

The differing regulatory environments of the EU, China, and US offer a natural experiment in spectrum-sharing approaches. As 6G approaches, stakeholders are examining which practices yield the best efficiency, fairness, and feasibility in sharing spectrum among many users and services.

5.1.5.1 Spectrum Utilization Efficiency

Efficiency refers to how well spectrum is used (minimizing idle frequencies and maximizing data throughputs). The US's CBRS model has shown high utilization gains: it unlocked underused federal spectrum and made it available to thousands of small-cell deployments [32]. Reports indicate that dynamic sharing can improve mid-band utilization dramatically for IoT applications [27]. Europe's local licensing also improves efficiency by localizing use – industries light up spectrum exactly where needed. However, LSA's initial static approach was less efficient in fast-changing scenarios (incumbent info was static, so spectrum might be left unused as a precaution). Incorporating real-time data (e.g. environmental sensing, geolocation analytics) is a best practice to boost efficiency. An emerging best practice is geographical spectrum reuse: Hong Kong's approach of issuing many micro-licenses of 100 MHz in 26/28 GHz across different 50 km^2 zones allows the same spectrum to be reused in multiple localities, a template 6G could emulate in dense cities for local subnetworks. In short, dynamic, location-based sharing (CBRS-style or micro licensing) tends to yield higher efficiency than nationwide exclusivity, as long as interference is managed.

5.1.5.2 Fairness and Market Innovation

Fairness in spectrum sharing includes giving diverse users (incumbents, large operators, small entrants, public and private entities) a chance to access spectrum resources. The CBRS GAA tier is cited as a proinnovation measure: it lowered the barrier for new players (enterprises, small ISPs, even individuals) to use prime spectrum without auction costs. This has spurred a new ecosystem of private network solutions in the US. Europe's local licensing is also fair in the sense of diversifying spectrum access – instead of only nationwide operators, now factories and research institutes can obtain spectrum rights. On the other hand, China's model currently scores low on fairness to non-operators: enterprises are entirely dependent on operators. While this ensured rapid nationwide 5G rollout (no fragmentation), it may stifle niche innovation by factories or IoT startups that in the US/EU could experiment with their own spectrum. A best practice emerging from EU/US experiences is tiered access with at least one license-exempt or lightly-licensed tier. This guarantees that innovators and underserved communities can tap into spectrum (for instance, rural WISPs using CBRS GAA to provide broadband). Fairness also relates to protecting incumbents' rights – over-prioritizing new usage could be seen as unfair to those who relied on spectrum (e.g. weather satellite operators fearing 5G in adjacent bands). So regulators strive for balance: in CBRS, incumbents have absolute priority (fair to them), while others share leftover capacity (fair to new users). Europe's LSA was fair to incumbents (they kept priority), but perhaps too fair – incumbents had little incentive to participate since they lost nothing by not sharing. The lesson learned is incentives matter: US offered incentives by not disrupting incumbents but allowing revenue from PAL auctions and new services; EU is now considering compensating incumbents or using mandate (e.g. if a band is underused, you must allow LSA). For 6G, global best practice likely means regulatory flexibility, which means enabling both exclusive and shared usage models and letting the market decide optimal mixes, under regulator's eye to ensure no one is anti-competitively hogging spectrum.

5.1.5.3 Feasibility and Complexity

A framework can be efficient and fair in theory but difficult to implement (feasibility). The CBRS SAS approach, while effective, required significant technical coordination – the FCC had to certify multiple SAS providers, define detailed protocols, and industry had to develop SAS-client software for base stations. It took years from concept to commercial launch. Europe's LSA also faced feasibility issues due to needing trust and data sharing between military and commercial entities. In contrast, simpler approaches like license-exempt have immediate feasibility, they piggyback on standards (Wi-Fi) and need only simple rules, though at the cost of no guarantees. The cooperative database model has proven feasible in the US and is being adopted in other contexts (Canada and others are considering CBRS-like models [36], and AFC for 6 GHz is global). A best practice is to start with a pilot in a manageable band: CBRS was piloted for several years with test users before scaling; regulators globally are now piloting local 5G spectrum (Germany's 3.7 GHz was a pilot for Industry 4.0). Another best practice is international knowledge-sharing: regulators share results of these pilots via groups like the ITU-R and WRC forums, so each region can adapt successful elements. For example, Europe can watch the US 6 GHz unlicensed deployment to decide how to manage 6 GHz for 6G; the US can observe Germany's private network success to inform some FCC rules easing experimental licenses for private 5G. Automation is also key to feasibility at 6G scale: with potentially millions of local subnetworks, manual coordination is impossible. So the use of AI for interference management, as academic studies suggest, will likely become best practice, albeit requiring new regulatory acceptance of algorithms making spectrum decisions.

5.1.5.3.1 Case Study – Industrial IoT

A manufacturing plant in Germany can now get a local 3.7 GHz license and deploy a 5G network with guaranteed spectrum. In the US, a similar plant could use CBRS GAA or acquire PAL licenses at auction, or use unlicensed Wi-Fi 6E. In China, the plant must partner with, say, China Unicom to set up a private slice. In practice, the German approach gives the enterprise full control and predictable performance (they have exclusive rights in their area), the US approach gives flexibility (no need to wait for a license if GAA is fine, but risk of potential interference from neighbours), and the Chinese approach gives high reliability via operator expertise but less flexibility. Measurements have shown that Germany's private networks achieve highly reliable low-latency links for industrial robots, but the ecosystem of devices is smaller (limited to certain vendor gear tuned to 3.7–3.8 GHz) [28]. The US CBRS networks have more device choice (since CBRS band is supported by many phones, IoT modules now), but GAA users occasionally face frequency shifts if a higher-tier user comes online. These differences impact how quickly in-X subnetworks can be deployed and scaled. Global best practice might be emerging as a hybrid: allocate some local licensing (for those needing strict reliability and willing to manage it) and have a general shared band (for quick, ad-hoc deployments). 6G could formalize this by designating, for example, a "6G industrial band" with a tiered access , a portion can be licensed locally, and a portion is open for dynamic access, all managed by a common framework.

To synthesize, Table 2 provides a structured comparison of the spectrum-sharing strategies and their impacts in the EU, China, and US, reflecting the above points.

Table 3 Comparative Overview of Spectrum-Sharing Strategies

Strategy / Feature	EU Perspective	China Perspective	US Perspective
Licensed Shared Access (2-tier)	EU-origin concept (ETSI LSA). Pioneered in 2.3 GHz; ensures QoS for secondary user; limited adoption so far . Likely revival for 6G (eLSA with dynamic features).	Not formally adopted. Spectrum sharing usually via internal arrangements (e.g. joint-operator use). LSA-like models possible in future if government opens specific bands to industrial use under license.	Superseded by 3-tier CBRS approach; FCC favors more dynamic, multi-user frameworks. (However, some 2-tier sharing exists: e.g. unlicensed users vs incumbents in 6 GHz with AFC is effectively 2-tier – incumbents and unlicensed.)
CBRS 3-Tier (Database-driven)	No direct equivalent yet. Discussions on adapting similar SAS model for other bands (e.g. 3.8–4.2 GHz or 6 GHz standard power) are ongoing. EU focuses on simpler two-tier for now, but 6G may necessitate multi-tier if multiple services cohabit a band.	No current 3-tier system. Policy is cautious – a SAS-like approach would likely be government-run. Could be considered for low-priority bands in future (but not evident yet).	Flagship sharing framework. Successfully operational for 4G/5G and IoT. U.S. exploring extending SAS to other bands (e.g. 3.1–3.45 GHz DoD spectrum). Proven model for balancing incumbent protection and new access.

Cooperative Sharing	High cooperation in LSA (regulated by license terms) and local licensing (coordination between regulator, incumbent, and new user). EU also encourages MNOs to lease spectrum to others (voluntary cooperation). Seen as necessary for critical QoS scenarios.	Very high reliance on cooperation *within* state apparatus (operators follow MIIT directives). If spectrum is to be shared, it's via top-down coordination (e.g. coordinated indoor use in shared band). Not much horizontal cooperation (enterprise to government directly).	Embraces cooperative systems (SAS, AFC). Industry consortia (OnGo Alliance for CBRS) exemplify cooperation between stakeholders and government. Cooperative sharing yields reliability (e.g. PALs get guarantees). Some push for even more collaborative frameworks (e.g. federal-commercial info sharing in real-time).
Non-Cooperative Sharing	Common in unlicensed bands (LBT, duty cycle enforcement by device design). EU mandates these etiquettes (per ETSI norms). Generally works well for moderate densities (e.g. Wi-Fi in offices), but industrial settings might face unpredictable performance if relying solely on unlicensed. EU balance: use unlicensed for supplementing capacity, not for ultra-critical links.	Default for unlicensed use – devices operate under general rules, no central coordination (just like elsewhere). In practice, fewer non-cooperative frameworks for big spectrum because China hasn't opened large unlicensed bands akin to US 6 GHz. Non-cooperative use mostly short-range IoT and Wi-Fi. For 6G, might maintain stricter control (preferring licensed or managed sharing for anything critical).	Huge ecosystem of non-cooperative sharing (Wi-Fi, Bluetooth, etc.). Tolerates higher interference risk in exchange for innovation (e.g. 2.4 GHz very crowded but has enabled IoT boom). The FCC sees unlicensed and cooperative sharing as complementary. Non-cooperative methods like DFS are integral to protect incumbents (radar) without direct coordination. Expect continuation of this approach (e.g. unlicensed use of mmWave and sub-THz will likely be non-cooperative with simple rules due to device count).

Dynamic Spectrum (Real-Time Adaptation)	Europe is researching AI-driven RRM for 6G subnetworks [2]. Some 5G features (e.g. dynamic spectrum sharing between 4G/5G) adopted by EU operators. Regulators open to dynamic ideas (e.g. temporary spectrum access for events). Likely to encourage dynamic local allocations via automated systems by 6G era.	Concept acknowledged by academics, but regulatory adoption slow. Any dynamic system would be tightly overseen by MIIT. Could leverage China's strength in AI – possibly a government AI system managing spectrum nationally in real time by 6G. For now, dynamic = operators optimizing their own spectrum use with tech like carrier aggregation.	The most dynamic regulatory practice globally. SAS and AFC are essentially real-time coordinators. FCC also has a system of experimental licenses that can be almost instantly approved for short-term use in various bands (e.g. for 6G trials). Going forward, US likely to continue pushing boundaries (maybe real-time auctions or dynamic marketplaces for spectrum are on the horizon as concepts).

Ultimately, each region's experience offers lessons. Europe's approach underscores the value of direct empowerment of verticals (through local spectrum access) and the importance of stable regulatory conditions for industrial investment. China's approach highlights the efficiency of centralized planning – it quickly achieved wide 5G coverage – but it may evolve to incorporate more sharing as 6G demands more localized innovation. The US approach shows that a mix of unlicensed and dynamic-licensed sharing can drive both innovation and efficient use, though it requires complex initial coordination. A likely global best practice for 6G is a multi-pronged spectrum strategy: exclusive licenses for wide-area coverage, shared or local licenses for industrial and specialized subnetworks, and unlicensed bands for ubiquitous low-cost connectivity. Such a diversified approach can maximize spectral efficiency while meeting the varied requirements of 6G use cases.

5.1.6 Summary

The regulatory landscape for spectrum sharing in in-X subnetworks, encompassing industrial, automotive, and IoT applications, varies significantly across different regions. While regions (US, China and EU) recognize the growing demand for flexible and efficient spectrum access in 6G networks, their approaches diverge based on existing policies, infrastructure, and industry structures. The EU adopts a hybrid approach, balancing exclusive mobile operator licenses with Licensed Shared Access and local industrial spectrum allocations, such as Germany's 3.7–3.8 GHz private 5G licenses. In contrast, China maintains a centralized model, where state-owned carriers retain control over all licensed spectrum, limiting enterprises to partnerships or network slicing arrangements. Meanwhile, the US leads in database-driven sharing, as exemplified by the Citizens Broadband Radio Service, which enables dynamic three-tiered spectrum allocation based on real-time demand.

In terms of spectrum-sharing techniques, LSA in the EU offers a structured two-tier sharing model, but adoption has been slow due to administrative complexities and limited incentives for incumbents. The CBRS framework in the US has demonstrated a more dynamic and efficient reuse of underutilized spectrum, ensuring fair access while protecting incumbents. China's co-sharing approach, particularly its 3300–3400 MHz band, suggests a gradual shift toward multi-operator spectrum allocation, albeit within a highly controlled regulatory environment. Across all regions, compliance mechanisms remain a critical component of spectrum-sharing policies. The US relies on automated enforcement tools, such as the SAS for CBRS and AFC for 6 GHz Wi-Fi, ensuring that lower-priority users do not interfere with incumbents. Europe and China, on the other hand, still operate under static or coordinated licensing models, though discussions on AI-driven dynamic spectrum management are gaining traction.

Despite these regulatory differences, several best practices are emerging. Tiered spectrum access models, such as CBRS, have been effective in improving utilization efficiency by dynamically allocating spectrum to secondary users without disrupting incumbents. Local or private spectrum licenses, as seen in the EU, provide enterprises with greater control over their industrial 6G networks, ensuring predictable performance in mission-critical applications. However, incentives for incumbents remain a key challenge, as regulators must find ways to encourage military, satellite, and other legacy users to release underutilized spectrum for shared use. Future spectrum considerations for 6G will likely center around mid-band extensions in the 7–15 GHz range, which offer greater bandwidth while retaining reasonable propagation characteristics. However, these bands are already occupied by incumbents such as satellite and radar services, necessitating carefully designed sharing mechanisms, possibly through LSA-type frameworks or real-time spectrum databases. In the sub-terahertz range (100–300 GHz), new regulatory models may be required to accommodate ultra-high-speed short-range communications, potentially allowing liberal unlicensed access due to the inherently limited propagation distance. Additionally, existing mmWave bands (24–86 GHz) will likely require refarming and dynamic allocation to optimize spectrum use for dense urban networks and automotive 6G applications.

5.2 GRANTED SUBNETWORK RESOURCE SHARING

5.2.1 Introduction

For UE-to-UE communication 3GPP today offers SL, where spectrum sharing takes place with the BS assigning specific licensed resources for use in SL [10]. Explicitly, SL focuses on device-to-device (D2D) communication avoiding the need for data to go through the BS. An advancement to that is SL relay, increasing the cellular coverage by enabling relaying data through SL [16]. In SL mode 1 the data is transmitted and received between devices directly, but the control remains solely within the BS, requiring BS interaction throughout the whole SL communication. SL UEs need to monitor PDCCH from BS, indicate SL BSRs to BS and get scheduled via the BS. This control overhead is resulting in higher power consumption at the UE side, higher latency due to the indirection of control via the BS and NW complexity. From user privacy perspective it also becomes a concern since all active communication links among UEs will be known by the BS. Finally, the BS involvement in every step, as in SL mode 1, makes the system less flexible for customization towards different UCs and may become a bottleneck in more dense deployments. In comparison to that SL Mode 2 [10] is relying on a higher degree of UE autonomy, but it is primarily defined for out-of-coverage scenarios and requires UEs to be pre-configured by the NW to operate in this mode.

5.2.2 Granted Subnetwork Resources

Instead, the BS shall delegate dedicated resources towards individual SNs in a more dynamic fashion to allow them to operate within these resources more freely and make use of the resources in an optimal way as shown in Figure 110. Those granted resources can be used by the MgtN for communication within the SN without further BS involvement. The MgtN may perform broadcast, multicast, unicast communication towards the SN and its devices and make use of the more short-range communication compared to UE-to-BS communication. The SN devices do not require to monitor PDCCH of the BS any longer for resource allocation as compared to SL mode 1. Altogether this leads to lower latency, less power consuming local communication within the SN.

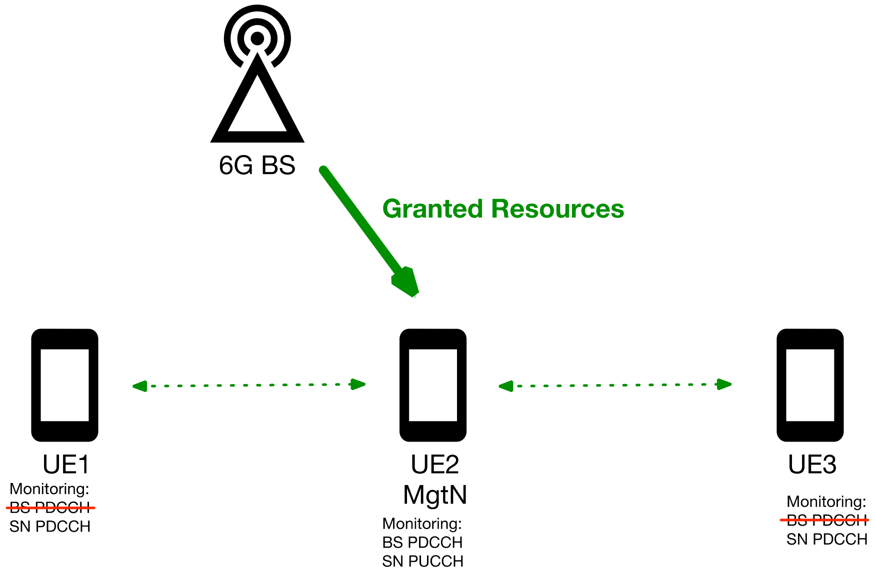


Figure 110 Granted Subnetwork Resources

Figure 111 shows the legacy SL Resource sharing concept from SL with Resource Pools (RP) within a SL BWP. This is a rather static scheme, where SL RPs are reserved for SL all the time in a common BWP for all UEs in the cell, configured via RRC [12].

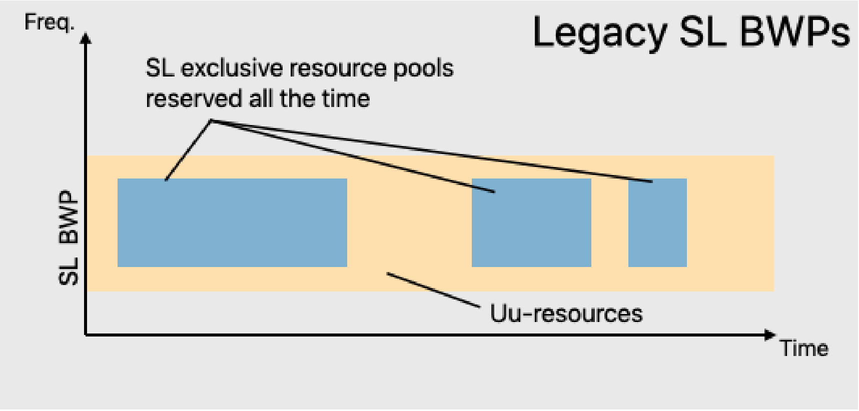


Figure 111 SL Resource Pools

To enable sharing of resources with multiple independent SNs, which are dynamic in their topology, size and QoS requirements, requires a more flexible scheme like shown in Figure 112. There newly introduced Exclusive Resource Pools (ERP) and Shared Resource Pools (SRP) are defined. As depicted those different types of resources can be used for different purposes, e.g. for SN Control Channels and SN Data Channels.

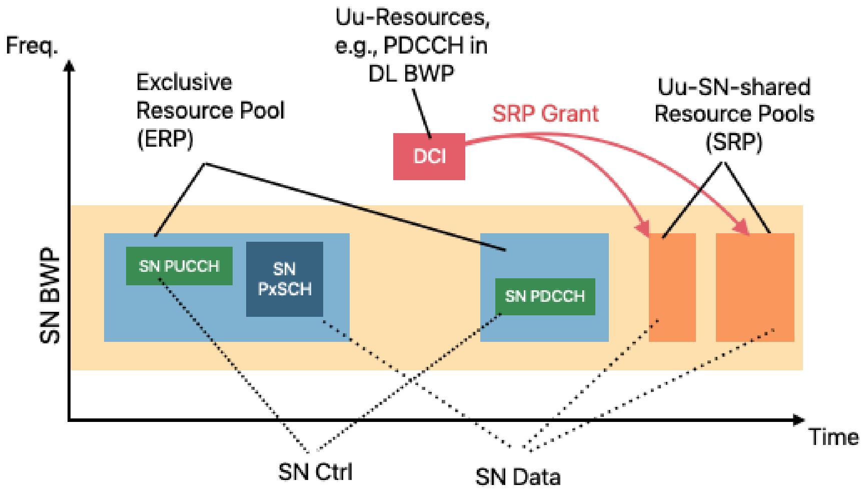

Freq.
Exclusive Resource Pool (ERP)
Uu-Resources, e.g., PDCCH in DL BWP
DCI
SRP Grant
Uu-SN-shared Resource Pools (SRP)
SN BWP
SN PUCCH
SN PxSCH
SN PDCCH
Time
SN Ctrl
SN Data

Figure 112 Dynamic Resource Grant via Exclusive and Shared Resource Pools

ERPs are RPs that are configured to SNs and rather static, they may be configured via RRC messaging and always usable by the SN for the basic communication. The ERPs are dedicated to individual SNs. The MgtN controls the access within the ERP for its own SN without UEs competing on them as in SL Mode 2. As shown in Figure 112 those ERPs can be used for SN-related control channels or channels that need robustness, like SN PDCCH/PUCCH or SN Discovery Channels. SRPs are only temporary and might not be exclusive per SN, they could be shared among different SNs or between Uu-traffic managed by the BS and SN-internal traffic managed by the MgtN, aka. *Uu-SN-shared*. The configuration of those SRPs is also shared via RRC messages, but in contrast to ERPs, the SRPs are granted by the NW more dynamically via the Uu-Interface to a particular SN, e.g. as an SRP Grant via DCI (see Figure 112) or MAC CE. This flexibility allows the BS to utilize the resources in an optimal way by controlling the balance between regular Uu-communication and SN-internal resource demands. If a cell is loaded, the BS may assign more SRPs for Uu-communication. When the cell load decreases the BS may indicate which SRPs can be utilized for SN-internal to certain SNs. None of the SRP resources are reserved in a fixed manner like in SL. Figure 113 shows an example of allocated ERPs and SRPs over time for a particular SN and how e.g. the size of the SN, reports from the MgtN or the NW conditions impact the amount of granted resources.

Figure 113 Dynamic SRP Grants over time

With the more flexible resource granting scheme the NW is able to control the balance between Uu-Traffic, SN-internal traffic. This control requires a few new metrics reported from the SN to determine the resources need at any given time, e.g.:

- **Number of active UEs in the SN**
    - NW is aware of the UEs joining or leaving the SNs as proposed in D4.2 [4], hence it is aware of the number of subscribers and their individual requirements and can adapt the granted resources accordingly, e.g.:
        - For best effort data UEs, only a few SRPs might be granted
        - For IMS/VoIP UEs, more SRP resources might be granted when calls are active
    - RRC Connected/Idle status of SN UEs define the “active size”, which can be reflected in
- **Maximum communication distance in the SN**
    - If communication is very localized (i.e., short distance) the amount of resources needed for SN-internal traffic could be less compared to larger distance communication, since the resources can be utilized more efficiently e.g., by using higher order modulation schemes.
- **SN Resource Demand**
    - Direct Resource Requests, e.g. in the form of SN Buffer Status Reports, accumulated by the MgtN and indicated as resource need to the NW
    - Note: This is providing more privacy for SN UEs by not revealing any information about communication links within the SN

Figure 114 Message sequence for SRP Grant

In Figure 114 the message flow is shown where the BS configures a SN with ERPs and SRPs in the beginning with the ERP being active right away. This is followed by granting SRPs for a certain duration or tight to conditions, e.g. serving cell quality, towards the SN. It shows serval alternatives on how and when the SRP Grant may be adapted based on changes in the NW or the SN, based on time or other conditions.

5.2.3 Summary

Overall, this scheme enhances RPs and provides more flexibility and adaptability compared to SL as the amount of resources reserved for SN is not fixed and BS can manage the balance between SN-resources and Uu-resources and among SNs dynamically. In particular, 6G SHINE D4.3 is elaborating how some portions of the licensed spectrum may even be better suited to opportunistic usage within subnetworks. Additionally, it enables more UC-centric optimizations, with respect to resource usage, latency or power due to the more independent resource management within the SN. The user privacy is increased compared to NW-centric solutions like SL, since SN-internal communication is hidden to the NW.

6 DISCUSSION TOWARDS 6G-SHINE OBJECTIVES

6G-SHINE has been developing new methods for a cost effective and constructive integration of subnetworks in the larger 6G network, where traffic, spectrum and computational load can be efficiently split between subnetworks and larger 6G networks, as well as solutions for efficiently managing the traffic types among subnetworks allocated in the same entity. These works are presented in this document and relate to project Objective 6.

More specifically in Objective 6 of the proposal, there was the commitment to develop new methods for integration of subnetworks in the 6G architecture and efficient orchestration of radio and computational resources among subnetworks and wider network. The motivation and scope rely upon the fact that 6G subnetworks must be able to operate autonomously, especially when eventual connection with a wider 6G network is intermittent or lost. Subnetworks shall be at the same time a component of the larger 6G 'network of networks', that offers opportunities for orchestration of their operations, and for improving their performance in terms of resource utilization, data traffic steering, management of spectrum and compute resources. Also, solutions for efficiently managing the traffic types among subnetworks allocated in the same entity have been developed.

For the assessment of these works in relation to Objective 6, it is important to note the novelty of the subnetwork concept. Many of the proposed functionalities are new while some are enhancements to the state of the art. The anticipated density of communicating nodes for the 6G area would make it unscalable to consider each communicating node as individual UE connected directly to a BS. The complexity and signalling at the NW side would inherently increase by that. Subnetworks with their means to have local management of communication and compute resources and their ability to coordinate for control and data plane functionalities will reduce this signalling and complexity burden on the NW. Hence the anticipated complexity and signalling increase in the NW can be mitigated thanks to the subnetworks, thus making the system more scalable.

For the assessment described below, it is also important to note the lack of existing procedures for baseline benchmarking. This gap in existing procedures creates the need for a qualitative assessment for some of the work proposed in this document. This is also true for some cases where procedures are already in place, and that are enhanced in this document - in some instances, the enhancement consists of leveraging the current protocol frameworks in the 3GPP system, introducing new information elements that are able to spread information efficiently across the end-to-end parent 6G network and all the nodes in the subnetworks. These enhancements alter substantially the behaviour of the 6G system, of the subnetworks, and their integration, without incurring in penalties in terms of signalling exchanges. In some other cases, where numerical results and assessment are provided, we discuss on the applicability of the protocol solutions proposed in this document that can be used to obtain such satisfactory results.

Adopting the structure of this document for discussion, Chapter 3 covered more fundamental aspects of the integration of subnetworks. These include subnetwork formation (including authentication aspects), connection of subnetwork and subnetwork nodes to the parent 6G network (their integration), measurements and mobility aspects, and QoS for the subnetwork. It stands out clearly that subnetwork formation and their connection are totally new aspects proposed, while measurements, mobility, and QoS are enhancements, as there are currently existing frameworks for those.

The proposed new type of NSA LC devices provides direct access to 6G Base Stations (BS) for low-latency communication. This enables reduced functionality devices to be deployed in use cases, such as the immersive education, requiring a higher degree of power efficiency as well as low latency (see Section 3.1). In addition, the ad-hoc formation of SNs and mobility of the involved devices are new aspects that need to be tackled in any of the SN use-cases (see Section 3.2). Along these lines, the establishment of mutual trust among those devices is very important aspect that needs to be considered and new procedures for mutual authentication of devices are proposed in 3.1.3. Once SNs are formed, the SN entities and the 6G BS need to ensure to route local traffic locally to achieve low latency end-to-end communication (3.1.4) as well as to provide seamless functional offloading within the SN (3.3.2) to enable power efficient XR devices for immersive education. With the XR-related traffic types the aspect of multi-modality among devices and within SNs become crucial and new procedures and scheduling methods are necessary as proposed in Section 3.4. Furthermore, aspects to also handle the delivery of data in a synchronized manner to get a true immersive experience for all involved parties in e.g. the gaming use case is highly relevant.

Chapter 4 discussed aspects related to the management of compute resources in the context of subnetworks integrated in a parent 6G network. Enabling compute offloading in the complete 6G-SHINE reference architecture was presented, including the design of new protocols for compute capability usage amongst all nodes, a new framework for quality of the compute service was introduced, and evaluation scenarios were considered and studied for the offloading of tasks, out of which numerical results were produced that validate proposed approaches and highlight the need to address compute resource sharing in an end to end manner.
Section 4.1 and 4.2 introduced a new framework for local and decentralized compute offloading to enable efficient orchestration of radio and computational resources within and among subnetworks.
In Section 4.5, CATS agents can be embedded in any subnetwork or 6G network compute nodes, and the proposed protocols focus on the information that needs to be exchanged to allocate compute tasks, taking into account compute and connectivity characteristics. The proposals are all based on request response type mechanisms, keeping the protocol design to the highest degree of simplicity.

The study in Section 4.3 demonstrated the effectiveness of a deterministic task offloading and resource allocation scheme, designed for the joint management of communication and computing resources across the IoT-edge-cloud continuum. The targeted KPIs include reliability, resource utilization efficiency, determinism under high transmission loads, and scalability with varying numbers of tasks and subnetworks. Our analysis showed that a deterministic approach to task scheduling can better guarantee deterministic service levels compared to state-of-the-art methods, increasing the ratio of satisfied tasks by up to 15%. The proposed scheme also ensures higher reliability, maintaining over 95% task reliability and achieving up to a 29% improvement over benchmark schemes. By flexibly managing task deadlines, the deterministic strategy achieves more balanced workload and resource distribution across the continuum, reducing resource saturation probability by up to 70%. Furthermore, it demonstrates significantly better scalability, supporting a larger number of tasks and subnetworks with up to a 75% improvement. These findings highlight the strong potential of the deterministic task offloading and resource allocation scheme in delivering bounded latency, high reliability, scalability, and efficient resource utilization.

Section 4.4 introduced a novel deterministic task scheduling scheme for in-vehicle networks and demonstrated its potential to leverage the capabilities of in-vehicle zonal E/E architectures with centralized computing. The targeted KPIs include latency, reliability, load distribution, and determinism under high transmission loads with diverse requirements. Our analysis showed that a deterministic approach for task scheduling can better guarantee deterministic service levels than state-of-the-art approaches and increase the ratio of satisfied tasks by up to 31%. In addition, the proposed deterministic task scheduling scheme can ensure higher reliability, supporting over 95% reliability for significantly more in-vehicle computational workloads achieving up to a 100% improvement compared to benchmark schemes. It also reduces resource saturation through more balanced workload distribution and resource utilization across the IVN. These trends have been validated across various IVN topologies, including configurations with wireless connectivity in hybrid IVN setups. The deterministic approach reduces latency in centralized mesh IVN topologies by up to 16.3% compared to other IVN topologies, representing a 95% to 210% improvement over benchmark schemes. These findings highlight the strong potential of the deterministic scheduling proposal for in-vehicle networks, ensuring bounded latency, high reliability, and efficient resource usage.

Chapter 5 dealt with spectrum usage in the context of integrated subnetworks in 6G. This is well aligned with the 6G-SHINE mission and is discussed here from the regulatory perspective for resource pooling.
A complete spectrum policy review was provided, highlighting the complexity of regulatory framework in different parts of the world, a state-of-the-art review of spectrum sharing techniques, and recommended best practices. This illustrates the difficulty of dealing with spectrum usage for subnetworks, that need to efficiently manage it across multiple nodes.
In addition, it describes how bulk resources granted by a 6G BS towards a SN provide a new, more flexible and adaptable mechanism compared to SL and enables more UC-centric optimization and customization with respect to resource usage, latency or power.

In conclusion, the new methods for integration of subnetworks in the 6G architecture and efficient orchestration of radio and computational resources among subnetworks and wider network are in line with Objective 6 of the 6G-SHINE proposal. Explicitly, Objective 6 is to “develop new methods for integration of subnetworks in the 6G architecture and efficient orchestration of radio and computational resources among subnetworks and wider network”. This objective has been verified by the applicability of the solutions in cases where the subnetworks scale up, i.e., a large number of subnetwork nodes require integration with the parent 6G network. Table 1 provides a summary of these solutions as well as their link to the most suitable use cases.

7 CONCLUSIONS

The final studies with respect to the management of traffic, computational and spectrum resources among subnetworks in the same entity, and between subnetworks and 6G network have been presented in this deliverable. These studies have built upon the architectural framework [3] as well as the initial studies of [4], a complete framework has been presented for the operation of SNs, while guaranteeing user privacy, autonomy and independence from the parent NW.

A new UE category, namely that of the NSA-UE, has been introduced in [4] and its procedural enhancements in terms of configuration, security and data multiplexing have been defined in Section 3.1.2. In terms of architectural enhancements, a UE-centric authentication device to device framework is proposed in Section 3.1.3, circumventing the need for accessing the CN. A complete formation, registration and mobility has been proposed in Section 3.2. This framework builds upon the distributed flexible snCP and snUP and the SN-TP and SN-RP proposed in [4]. An extension to SN-RP enabling local IP routing has also been proposed in Section 3.1.4. In the context of coordination within SNs, CP offloading processes designed for location updates have been introduced in Section 3.3.2. Coordination has also been extended beyond the SN boundaries, among neighbouring SNs. The studies on SN-assisted predictive mobility of Section 3.2.4 and on the coordinated L3 measurement framework of Section 3.3.3 utilises this cross-SN coordination to improve the performance of the individual SN UEs. Additionally, the latter studies have demonstrated SN potency in improving the individual UEs capabilities and performance.

The aspect of data multi-modality has also been investigated in Section 3.4 along with its impact on QoS. Two approaches have been proposed for focusing on the consumer use case category [2]. Both approaches align packets in time that are belonging to different interrelated flows, commonly known as multi-modality flows. The first approach in Section 3.4.2 relies upon the existing 3GPP framework for the SL-Relay and time alignment is accomplished by adding information of the related flows, and of the interrelated packets in the related flows, in the packet headers. Thereby the relays and devices can synchronize the dataflows and still maintain a relevant packet delay budget which improves the performance and capacity of the network. The second approach in Section 3.4.4 follows a more UE-centric approach, moving time alignment control at the MgtN side. A novel DGF is introduced at the MgtN side, tasked with performing scheduling and data time alignment within the SN. This approach enables QoS even for local multi-modal traffic that does not leave the SN. Furthermore, UL scheduling enhancements have been presented in Section 3.4.3, where a per-UE BSR is introduced enabling the parent 6G NW to schedule SN UEs efficiently.

Moving on to compute offloading, this report continues to build upon the framework laid by [4], where the node roles necessary for enabling local compute offloading were introduced. In Section 4.1, an extension of these roles has been made for the sake of enabling decentralised compute offloading across SNs. Additionally, the methods necessary for enabling both local within the SN and decentralised compute offloading have been defined. These methods include among others the selection of CCNs as well that of connecting the ONs with their respective CompNs.

Achieving a converged computation and communication SN requires a revisit of the existing QoS framework to include computation aspects. This is necessary for fully utilizing the computation offloading capability in the SN architecture and to support computation requests with different resources and performance requirements. For this reason, a novel QoS framework has been introduced in Section 4.2, which supports both communication and computation within a SN, between SNs, and between the SN and the parent 6G NW. New SN QoCS parameters and characteristics to fulfil the required computation requirements along with the high-level procedures to support SN QoCS for local SN and decentralised compute offload have been presented.

For the vehicular use case category [2] in particular, the topology is rather static. Nevertheless, extreme reliability requirements arise due to the system being safety-critical. In this context, a study is presented in Section 4.3, which proposes a deterministic task offloading and resource allocation scheme for the integrated management of communication and computing resources across the IoT-edge-cloud continuum. This approach emphasizes prioritizing task deadlines over merely minimizing individual task execution latency, thereby ensuring a more efficient and balanced distribution of workloads throughout the continuum. By dynamically managing task completion deadlines, the deterministic strategy can more effectively adapt to varying operational conditions. Furthermore, the study highlights how this approach of performing task offloading and resource allocation can significantly enhance scalability in the next-generation cellular networks. An additional study is presented in Section 4.4 which introduces a novel deterministic task scheduling scheme for IVNs and demonstrates its potential to leverage the capabilities of in-vehicle zonal E/E architectures with centralized computing. The deterministic approach to task scheduling is shown to provide more reliable service levels compared to alternative methods, effectively supporting the growing computational workloads and tasks within the vehicle. This is accomplished through a more balanced distribution of workloads and optimized resource utilization across the IVN. These findings will be validated across a range of IVN topologies, including scenarios that incorporate wireless connectivity in hybrid IVN configurations.

To cope with the needs of AR/VR/XR related use cases, a CATS framework that enables joint compute and network-aware traffic steering was presented in section 4.5, relying on two key functional entities, the CATS agent and the CATS controller. This technique enables service instance selection and traffic steering by dynamically selecting the best service site (e.g., LC or HC nodes) based on real-time connectivity and computing constraints. Proposed work also addresses mobility-aware service anchoring and migration ensures seamless service continuity for mobile terminals in subnetworks. Generally speaking, QoE and QoS are improved by ensuring optimal service execution in heterogeneous, resource-constrained, and mobility-prone environments.

Moving on to the studies on dynamic spectrum sharing, they have been presented in Chapter 5. A review of the spectrum sharing regulations across countries is made in Section 5.1.3. More specifically, a comparison of licensed and license-exempt spectrum policies in the EU, China, and the US has been presented, followed by an evaluation of sharing mechanisms, such as EU's LSA, US CBRS models. Subsequently, a review of the compliance and enforcement approaches has been made in Section 5.1.4 leading to identification of the emerging trends in spectrum regulation. An analysis of the implications for future 6G spectrum policy have also been made in Section 5.1.5. Additionally, a novel protocol for flexible access of licensed resources has been introduced in Section 5.2. This framework enables the parent 6G NW to dynamically assign licenced spectral resources to registered SNs. For this reason, the concept of dynamic resource pools has been introduced, allowing the NW to assign them to SNs depending on their individual traffic needs.

Last but not least, the connection of the aforementioned methods to the objectives and targets of 6G-SHINE has been presented in Chapter 6.

REFERENCES

[1] B. Priyanto et al., “D2.1. – Initial Definition of Scenarios, Use Cases and Service Requirements for in-X Subnetworks,” 6G-SHINE, August 2023.

[2] B. Priyanto et al., “D2.2 – Refined Definition of Scenarios, Use Cases and Service Requirements for in-X Subnetworks,” 6G-SHINE, February 2024.

[3] P. Maia de Sant Ana *et al.*, “D2.4 - In-X subnetwork architectures and integration into 6G ‘networks of networks’”, 6G-SHINE, February 2025.

[4] D. Alanis et al., "D4.2 - Preliminary results on the management of traffic, computational and spectrum resources among subnetworks in the same entity, and between subnetworks and 6G network” 6G SHINE, June 2024.

[5] M. A. Uusitalo *et al*., "6G Vision, Value, Use Cases and Technologies From European 6G Flagship Project Hexa-X," in *IEEE Access,* vol. 9, pp. 160004-160020, 2021.

[6] M. Ericson *et al.*,” Deliverable D3.2 Initial Architectural enablers,” Hexa-X-II, October 2023.

[7] O. Akgul *et al.*, ”Deliverable D3.5 Final architectural framework and analysis,” Hexa-X-II, February 2025.

[8] 3GPP TS 33.501 v18.5.0, “Security architecture and procedures for 5G system”, May 2024

[9] 3GPP TS 23.304 v18.5.0, "Proximity based Services (ProSe) in the 5G System (5GS)”, March 2024

[10] 3GPP TS 38.300 v18.0.0, “NR; NR and NG-RAN Overall Description; Stage 2”, December 2023

[11] 3GPP TS 38.214 v18.0.0, “5G; NR; Physical layer procedures for data”, December 2023

[12] 3GPP TS 38.331 v18.2.0, “5G; NR; Radio Resource Control (RRC); Protocol specification”, August 2024

[13] 3GPP TS 38.213 v18.2.0, “5G; NR; Physical layer procedures for control”, May 2024

[14] 3GPP TS 38.321 v18.2.0, “5G; NR; Medium Access Control (MAC) protocol specification”, August 2024

[15] 3GPP TS 22.847 v18.1.0, “Study on supporting tactile and multi-modality communication services” December 2021.

[16] 3GPP TR 38.836 V2.0.0, “Study on NR sidelink relay”, March 2021

[17] 3GPP TS 37.324 v1.0.0, “NR; Service Data Adaptation Protocol (SDAP) specification”, May 2022

[18] 3GPP TS 23.501 v18.4.0, “System architecture for the 5G System (5GS); Stage 2”, December 2023

[19] 3GPP TS 38.300 v15.19.0, “NR; NR and NG-RAN Overall Description; Stage 2”, December 2024

[20] 3GPP TS 38.300 v16.18.0, “NR; NR and NG-RAN Overall Description; Stage 2”, December 2024

[21] Apple, R2-1909862, “Consecutive Conditional Handover,” 3GPP TSG-RAN WG2 Meeting #107, Prague, August 2019.

[22] MediaTek, RP-213565, “New WID on Further NR mobility enhancements,” 3GPP TSG RAN Meeting #94e, December 2021.

[23] S. Eldessoki *et al.*, “Distributed Compute Offloading in Local Communications,” in Proceedings of 2025 International Conference on Computing, Networking and Communications (ICNC), pp. 1-6, February 2025.

[24] 3GPP TS 23.287 v18.4.0, “5G; Architecture enhancements for 5G System (5GS) to support Vehicle-to-Everything (V2X) services,” October 2024.

[25] Apple, R2-2406664, “RRM measurement prediction results for case B,” 3GPP TSG-RAN WG2 Meeting #127, August 2024.

[26] Apple, R2-2408556, “RRM measurement prediction results using field and simulated data,” 3GPP TSG-RAN WG2 Meeting #127bis, October 2024.

[27] Brown, Colin, and Bo Rong. "Enhanced IoT Spectrum Utilization: Integrating Geospatial and Environmental Data for Advanced Mid-Band Spectrum Sharing." Sensors 24.18 (2024).

[28] ITU Digital Regulation Platform, "Spectrum licensing: local and private networks in Germany", 2020. Link available at: link.

[29] Dean Bubley. "Disruptive Wireless: Thought-leading wireless industry analysis". 2023. Link available at: link

[30] ITU Digital Regulation Platform, "Use of shared spectrum at the national level". 2020. Link available at: link

[31] Mike Dano, "6GHz, satellites and 6G addressed at WRC-23". 2023. Link available at: link

[32] Michael Woodley, "Is CBRS for everybody? – growing pains and progress towards a practical solution". 2022. Link available at: link

[33] Robert Clark, "China's 5G private networks slow in the Year of the Ox". 2021. Link available at: link

[34] Medeisis, Arturas, Vladislav Fomin, and William Webb. "Untangling the paradox of licensed shared access: Need for regulatory refocus." *Telecommunications Policy* 46.8 (2022).

[35] National Telecommunications and Information Administration. "Advanced Dynamic Spectrum Sharing Demonstration in the National Spectrum Strategy". 2024. Link Available at: link

[36] Rajeesh Radhakrishnan , "What are the International Differences in Private Networks?". 2023. Link Available at: link

[37] U. Mikko, et al. "European Vision for the 6G Network Ecosystem", *6G-IA Vision Working Group' White Paper*, Nov. 2024.

[38] B. Priyanto, et al. "6G-SHINE D2.2: Refined definition of scenarios use cases and service requirements for in-X subnetworks," Feb. 2024.

[39] S. Kerboeuf et al., "Design Methodology for 6G End-to-End System: Hexa-X-II Perspective", *IEEE Open Journal of the Communications Society, vol. 5, pp. 3368-3394*, May 2024.

[40] Smart Networks in the Context of NGI, Technical Annex to Strategic Research and Innovation Agenda 2022-27, pp. 1–358, Technical Annex to SRIA 2024, v0.31 for consultation, 2024. Technical Annex to SRIA 2024 v0.31 for consultation.pdf.

[41] G. P. Sharma *et al.*, "Toward deterministic communications in 6G networks: state of the art, open challenges and the way forward," *IEEE Access*, vol. 11, pp. 106898–106923, 2023.

[42] E.A. Vitucci, "6G-SHINE D2.3: Radio propagation characteristics for in-X subnetworks", Dec. 2024.

[43] 3GPP *TS 36.211 v18.0.1 (2024), Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation* (Release 18).

[44] Bicket, John Charles. "Bit-rate selection in wireless networks." *PhD diss., Massachusetts Institute of Technology*, 2005.

[45] B. H. Arabi, "Solving NP-complete Problems Using Genetic Algorithms", in *Proc. IEEE UKSim, pp. 43-48*, Cambridge, UK, 2016.

[46] Intel, "Case Study of Scaled-Up SKT 5G MEC Reference Architecture", *White Paper*, 2022.

[47] W. Fan *et al.*, "Joint task offloading and resource allocation for multi-access edge computing assisted by parked and moving vehicles," *IEEE Trans. Veh. Technol.*, vol. 71, no. 5, pp. 5314–5330, 2022.

[48] Hu, M., *et al.* "Heterogeneous edge offloading with incomplete information: A minority game approach," *IEEE Trans. Parallel Distrib. Syst.* vol. 31, no. 9, pp. 2139-2154, 2020.

[49] W. Fan *et al.*, "Joint task offloading and resource allocation for vehicular edge computing based on V2I and V2V modes," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 4, pp. 4277–4292, 2023.

[50] J. Cai *et al.*, "Multitask multi objective deep reinforcement learning-based task offloading method for industrial Internet of Things," *IEEE Internet Things J.*, vol. 10, no. 2, pp. 1848–1859, 2023.

[51] W. Feng *et al.*, “Latency minimization of reverse offloading in vehicular edge computing,” *IEEE Trans. Veh. Technol.*, vol. 71, no. 5, pp. 5343–5357, 2022.

[52] L. T. Oliveira *et al.*, “Enhancing modular application placement in a hierarchical fog computing: A latency and communication cost-sensitive approach,” *Comput. Commun.*, vol. 216, pp. 95–111, 2024.

[53] S. Hakimi, et al., "Rate-conforming Sub-band Allocation for In-factory Subnetworks: A Deep Neural Network Approach", *in Proc. EuCNC/6G Summit, pp. 729-734*, Antwerp (Belgium), July 2024.

[54] N. N. Dao, et al., "Self-Calibrated Edge Computation for Unmodeled Time-Sensitive IoT Offloading Traffic" *IEEE Access, vol. 8, pp. 110316-110323*, June 2020.

[55] 3GPP TS22.104 v19.2.0 (2024), “Service requirements for cyber-physical control applications in vertical domains (Release 19)”.

[56] R. K. Jain, D.-M. W. Chiu, W. R. Hawe, et al., “A quantitative measure of fairness and discrimination”, *Eastern Research Laboratory, Digital Equipment Corporation*, Hudson, MA, vol. 21, Sep. 1984.

[57] V. Bandur, et al., "Making the Case for Centralized Automotive E/E Architectures", *IEEE Trans. Veh. Technol., vol. 70, no. 2, pp. 1230-1245*, Feb. 2021.

[58] P. Laclau, et al., "Enhancing Automotive User Experience with Dynamic Service Orchestration for Software Defined Vehicles", *IEEE Trans. on Intell. Transp. Syst., vol. 26, no. 1, pp. 824-834*, Jan. 2025

[59] Robert BoschGmbH, “The next step in E/E architectures”, Aug. 2023. Accessed: Mar. 2025. [Online]. Available: https://www.bosch-mobility.com/en/mobility-topics/ee-architecture/

[60] A. Berisa, et al., “AVB-aware Routing and Scheduling for Critical Traffic in Time-sensitive Networks with Preemption”, *Proc. ACM 30th RTNS, pp. 207-2018*, Paris (France), 7-8 June 2022.

[61] S.D. McLean, et al., “Configuring ADAS platforms for automotive applications using metaheuristics”, *Front. Robot. AI, vol. 8*, Jan. 2022.

[62] B. Xu, et al., “A Joint Routing and Time-Slot Scheduling Load Balancing Algorithm for In-Vehicle TSN”, *IEEE Trans. Consum. Electron.* (early access on IEEE Xplore since Feb. 2025).

[63] Advantech, "ECU-4784," Accessed: Mar. 2025. [Online]. Available: https://www.advantech.com/en-us/products/1-369nwl/ecu-4784/mod_18553282-e8f5-4b32-a64b-1083f7182d36.

[64] Y. Deng, et al., “Multi-hop task routing in vehicle-assisted collaborative edge computing”, *IEEE Trans. Veh. Technol.*, vol. 73, no. 2, pp. 2444–2455, 2023.

[65] E.A. Vitucci, et al. “6G-SHINE D2.3: Radio propagation characteristics for in-X subnetworks”, Dec. 2024.

[66] RP-243318, Revised WID on XR (eXtended Reality) for NR Phase 3

[67] ITU-R M.2160-0, “Framework and overall objectives of the future development of IMT for 2030 and beyond