Project number: 826276

CPS4EU

Cyber Physical Systems for Europe

D2.4 - Propositions for 5G, including URLLC evolution – v2

Reviewer (name – company): V. Savin (CEA), M. Maman (CEA), J.B. Doré (CEA), J. Schmitt (VSORA), D. Choukroun (SEQUANS)

Dissemination level: Public

Version	Date	Author (name – company)	Comments
V1.0	2022/08/31	R. Bertolini (CEA) S. Bicaïs (CEA) D. Demmer (CEA) J.B. Doré (CEA) A. Falempin (CEA) M. Maman (CEA) M. Sana (CEA) V. Savin (CEA) J. Schmitt (VSORA) T. D. Nguyen (VSORA)	

EXECUTIVE SUMMARY

In recent years, Cyber Physical System (CPS) technologies have become a game changer in strategic sectors where Europe is a world leader, such as Automotive, Energy and Industry Automation. CPS4EU is a European project that aims at developing enabling technologies for CPSs, encompassing various vertical use cases. CPS4EU's main overall goal is to strengthen the CPS value chain by creating world-class European Small and Medium Enterprises (SME) and by providing CPS technologies that in turn will sustain the leadership of the large European groups in key economy sectors. In this way, the project is stimulating the development of innovative products to support the massive digitalization increasingly integrated in our everyday environment.

Work Package (WP) 2 of CPS4EU deals with the communication aspects of CPSs. It investigates how a CPS can be connected efficiently and securely to a network, in the context of emerging technologies, as Internet of Things (IoT) and 5G. It also targets the design and development of solutions that guarantee specific performance and quality of service for various CPS applications. The numerous emerging CPS application scenarios and use cases impose constraints and require features that cannot be met by nowadays technologies. Therefore, Task 2.2 of WP2 has the role of investigating novel communication solutions that surpass the state of the art and are especially adapted to the requirements coming from other WPs, analysed during the work of Task 2.1, and reported in deliverables [D2.1] and [D2.2].

As a result, the activities of Task 2.2 were devoted to the study of novel telecommunication solutions at different network layers and for different application scenarios, though all with the common goal of serving CPSs. They covered several aspects of CPS applications, such as network architectural enablers, innovations at the physical layer, resource orchestration solutions applied to edge computing scenarios, mobility aspects. A particular attention has been paid to enablers of Ultra-Reliable Low-Latency Communications (URLLC) and Time-Sensitive Networking (TSN), identified by the project as key features of an effective CPS-supporting communication infrastructure. The research work carried out was made of both theoretical analysis and numerical validation of the proposed solutions. Moreover, part of solutions developed in Task 2.2 have been selected for further development and evaluation in Task 2.3, as reported in deliverables [D2.5] and [D2.6].

The activities of Task 2.2 during the first half of the project's lifetime have been reported in deliverable [D2.3]. This deliverable [D2.4] is the second report of Task 2.2 and describes the activities of this task during the second half of the project's lifetime.

This deliverable is organized as follows. Section 1 briefly recalls the requirements and motivations identified within the Task 2.1 of WP2, and discusses the context and the relation between the project goals and the proposed studies. Section 2 summarizes the studies carried out in Task 2.2, during the entire lifetime of the project. While in line with the actual and foreseen standardization directions for modern communication networks, our work also makes a further step "beyond 5G" to address the new challenges posed by CPSs. These studies span through several domains: physical-layer enablers for URLLC networks at the service of CPSs, resource scheduling optimization for URLLC networks, optimization of collaborative communications in industrial environments, algorithms for energy efficiency in scenarios where CPSs leverage edge computing under latency and reliability constraints, TSN-dedicated network architectures, solutions for improving the reliability of the forward error correction scheme, and solutions for the improvement of the quality of wireless communications in mobility scenarios. The activities carried out during the second half of the project's lifetime are then further detailed in Section 3 of this deliverable. Finally, a general conclusion of the work carried out in Task 2.2 is given in Section 4.

Table of content

Acronyms

Acronym / abbreviation	Description
3GPP	Third Generation Partnership Project
4G	4th Generation of technology standard for broadband cellular networks
5G	5th Generation of technology standard for broadband cellular networks
6G	6th Generation of technology standard for broadband cellular networks
ACK	Acknowledgement
ALU	Arithmetic Logic Unit
AoA	Angle of Arrival
AoD	Angle of Departure
AP	Access Point
API	Application Programming Interface
AWGN	Additive White Gaussian Noise
BCH	Bose, Ray-Chaudhuri and Hocquenghem (code)
BER	Bit Error Rate
BP	Belief Propagation
BS	Base Station
BSR	Buffer Status Report
CBRA	Contention Based Random Access
CFO	Carrier Frequency Offset
CFRA	Contention Free Random Access
CN	Core Network
COTS	Commercial Off The Shelf
CPS4EU	Cyber-Physical Systems for Europe
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CRS	Cell-specific Reference Signal
CSI	Channel State Information
CU	Centralized Unit
DCI	Downlink Control Information
DL	Downlink
D-MEC	Discontinuous Mobile Edge Computing
DRB	Data Radio Bearer
DSP	Digital Signal Processing
ED	Energy Detection
eNB	eNodeB
EPA	3GPP Extended Pedestrian A channel
EPC	Evolved Packet Core
FDD	Frequency-Division Duplex
FEC	Forward Error Correcting
FER	Frame Error Rate
FF	Feed-Forward
gNB	gNodeB
GNSS	Global Navigation Satellite Systems
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HARQ	Hybrid Automatic Repeat reQuest
HDPC	High Density Parity Check
HO	HandOver
HSS	Home Subscriber Server
HW	Hardware
ICT	Information and Communications Technology

IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IP	Internet Protocol
IPv4	Internet Protocol version 4
KPI	Key Parameter Indicator
LDPC	Low-Density Parity-Check
LLR	Log Likelihood Ratio
LOS	Line Of Sight
LTE	Long Term Evolution
MAC	Medium Access Control
MARL	Multi-Agent Reinforcement Learning
MCS	Modulation and Coding Scheme
MEC	Multi-access Edge Computing
MDP	Markov Decision Process
MIB	Master Information Block
MIMO	Multiple-input Multiple-output
ML	Maximum Likelihood
MME	Mobility Management Entity
mmWave	Millimetre Wave
NACK	Non-Acknowledgement
NAS	Non Access Stratum
NB-IoT	Narrow-Band Internet of Things
NLOS	Non Line Of Sight
NN	Neural-Network
NND	Neural-Network-based Detector
NR	New Radio
NSA	Non-StandAlone
OAI	Open Air Interface
OOK	On-Off Keying
PCRF	Policy and Charging Rules Function
PDCCH	Physical Downlink Control Channel
PDSCH	Physical Downlink Shared Channel
POMDP	Partially Observable Markov Decision Process
PPO	Proximal Policy Optimization
PRACH	Physical Random Access Channel
PSS	Primary Synchronization Signal
PUCCH	Physical Uplink Control Channel
QoS	Quality of Service
RA	Random Access
RAN	Radio Access Network
ReLU	Rectified Linear Unit
reTx	Retransmission
RF	Radio Frequency
RNN	Recurrent Neural Network
RRC	Radio Resource Control
RSRP	Reference Signal Received Power
RTL	Register-Transfer Level
RU	Radio Unit
Rx	Reception
SDN	Software-Defined Networking
SDR	Software-Defined Radio
SIB	System Information Block
SINR	Signal-to-Interference-Noise Ratio
SME	Small and Medium Enterprises
SNR	Signal-to-Noise Ratio
SPGW	Serving Gateway and Packet Data Network Gateway

This project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 826276

SRB	Signaling Radio Bearer
SSB	Synchronization Signal Block
SSS	Secondary Synchronization Signal
SW	Software
TDD	Time-Division Duplexing
TSN	Time Sensitive Networking
TTT	Time To Trigger
Tx	Transmission
UCI	Uplink Control Information
UDP	User Datagram Protocol
UE	User Equipment
UL	Uplink
ULSCH	Uplink Shared Channel
URLLC	Ultra-Reliable and Low-Latency Communications
USRP	Universal Software Radio Peripheral
V2X	Vehicle-to-everything
WP	Work Package

1. INTRODUCTION

1.1. Requirements and Motivation

The activities carried out in Task 2.2 are grounded on the requirements, use cases, and motivations collected in Task 2.1 from the vertical WPs with respect to the communication needs. The outcome of Task 2.1 was captured initially in deliverable [D2.1] and completed in deliverable [D2.2].

CPS4EU covers a large variety of vertical use cases, from energy, factory automation, automotive and we realized that every specific use case has finally its own set of requirements for communication. First, most of use cases mentioned communication as a “must have”. Very few consider it as just “nice to have”: for instance, automotive driving relies on local sensors and processing for emergency situations, but leverages communication for a better service (e.g. for map upgrade, traffic information retrieval, or simply secure software upgrade). This variety of use cases yield a variety of requirements with respect to communication: either very low and infrequent data exchanges (e.g. for sensors reporting few measurements per day like metering use cases) or more frequent (e.g. for asset tracking), and in many cases including very stringent requirements either in terms of data rate or in terms of latency or reliability (e.g. for remote control of moving devices, virtual or augmented reality, digital twin, coordinated lifting for cranes, robots coordination in factory 4.0, etc.).

Task 2.1 conducted a quick survey on existing technologies and we concluded that there is no single wireless communication technique that could meet all the requirements. However, we also realized that the cellular framework offers a set of solutions that could cover most of the requirements. Indeed, cat-M and NB-IoT is an efficient solution for battery powered low-end devices while LTE could meet higher-end use cases, thanks to its various categories (from category 1 allowing throughput of few tens of Mbps to category 18, up to around 1 Gbps). So, 4G is already a satisfactory candidate as a universal communication system for CPS.

However, some specific use case cannot be met with 4G mostly because of three characteristics: the need of low (and sometimes deterministic) latency; the need of higher throughput, and the foreseen explosion of the number of connected low-end devices (massive machine-type communications). 5G then becomes a natural candidate to fulfil such CPS requirements, especially for the significant flexibility it will provide, at least from the network perspective and deployment possibilities. Hence, with a single network solution, an operator could offer various “slices” with different QoS profiles. In addition, 5G extends the possibility to offer private network deployments: a network could be deployed specifically for a vertical, locally, e.g. over its factory, independently of consumer networks, naturally fulfilling requirements in terms of security and data protection (e.g. the network servers being hosted locally at the factory).

As a result, 5G opens the door for a new set of IoT, often referred to as “critical” or “industrial” IoT, encompassing the complete set of communication requirements of CPSs, as gathered from WP6, WP7, WP8 and WP9 of CPS4EU.

However, 5G was initially defined for the consumer market and the requirements of verticals were captured but not prioritized. The first 5G version, release 15, was mostly serving higher data rates, with little work on critical communications. Release 16 compensated a bit this gap but still important aspects are not fully defined, such as the notion of URLLC profile (what is exactly an URLLC communication?), how to integrate TSN within a 5G network, or how we could use new frequency bands, especially the millimetre-wave (mmWave) one, to support harsh communications scenarios.

In the perspective of 5G development and evolution, WP2 has also the mission to propose innovative solutions, in line with standard work, and to provide missing building blocks (it has to be noted that a standard ensures interoperability between device and network equipment, but does not define internal algorithms to support this interoperability). As a result, in WP2, Task 2.2 has the objective to work mid- to long-term solutions to enable the few CPS use cases that cannot be served with nowadays and emerging wireless communications systems.

1.2. Context and relation between the project goals and the proposed studies

As introduced above, CPSs represent key drivers for the innovation capacity of European industries in sectors like automotive, energy and industry automation. CPSs combine intensive connectivity, embedded computing and local intelligence to create a link between physical and digital worlds and to enable cooperation among systems. The importance of CPSs is increasing with massive digitalization and their development is opening new market opportunities and requires non-straightforward technological innovations. Therefore, one of CPS4EU’s main objective is to foster the creation of world-class CPS technology providers. In line with this goal, WP2 focuses on the design and development of CPS HW/SW communication modules to be used across different industrial sectors, leveraging requirements coming directly from the significant number of CPS4EU industrial partners, thus reinforcing their leadership in a very competitive market. The goal is to guarantee dynamic management of applicative flows (ex: reliability, latency, and data rate) and continuity of service in order to ensure the control, localization of vehicles and CPS devices, distribution and collection of contents and critical or non-critical data.

Connectivity is critical for CPSs. Indeed, it must bear communications of hundreds, or even thousands, of dynamically connected objects that are also connected to external operators while ensuring reliability. Such connectivity, whether object-to-object or object-to-network infrastructure, often needs to be wireless (or to integrate complementary wireless and wired features) to offer more flexibility, adaptability to changing environment and much lower cost of operation compared to a wired communication approach. It must also be able to auto-adapt dynamically in time and space and to support the differentiation of operations or services. Thus, CPS requirements in terms of connectivity introduce the need for a very limited latency of communication between the device and the infrastructure. This is particularly the case of remote control of heavy machinery in hazardous places or used for monitoring and controlling smart grids. The challenge is to provide extremely fast and reliable connectivity at the radio level, as well as for end-to-end performance of the system. This requires also a more distributed infrastructure design that include mobile edge computing (MEC) capabilities.

In general, to achieve the performance required by URLLC applications for CPSs, it is necessary to combine in the architecture of 5G networks a series of enablers and technological innovations. In particular, it will be necessary to implement at all layers solutions that make the network more robust and able to guarantee both low latencies and low transmission error probabilities. These solutions include multi-connectivity enablers (network reliability through spatial and frequency redundancy or diversity), high coordination and synchronization of network nodes (for instance via TSN) and new coding, modulation, and scheduling schemes for the PHY, MAC, and transport layers.

One of the goal of CPS4EU's WP2 is to investigate these URLLC enablers and propose novel technological and theoretical solutions for 5G networks and beyond, adapted to CPS-specific use cases and applications. These solutions may be defined on top or be complementary to the solutions defined by the standardisation bodies addressing these challenges, such as 3GPP or IEEE. Accordingly, the activities carried out in Task 2.2 cover several aspects of URLLC and CPS applications, such as network architectural enablers, innovations at the physical layer, resource orchestration solutions for URLLC applied to edge computing scenarios, mobility aspects. These activities are summarized in the next section.

2. SUMMARY OF THE CONTRIBUTIONS WITHIN TASK 2.2

In this section, we provide a global overview of the activities carried out in Task 2.2 during the entire lifetime of the project. We summarize first the activities carried out during the first half of the project (Section 2.1), reported in deliverable [D2.3], and then the activities carried out during the second half of the project (Section 2.2). To avoid content duplication, only the activities carried out during the second half of the project are further detailed in Section 3 of this document.

2.1. Summary of the activities carried out during the first half of the project

Design and analysis of MIMO systems using energy detectors for mmWave applications (D2.3, Section 3.2). A common feature of CPSs, is the densification of the network and the presence of more access points and more cooperation between them, typically in “mesh” topologies. To guarantee URLLC services, it is therefore necessary to make communication between access points and backhauling extremely efficient, even more so than they already are. Therefore, in D2.3, we presented our innovative results on mmWave MIMO xHauling. The demand for wireless mmWave xHaul connection is increasing especially for the rapid deployment of private networks. mmWave xHauls improve the data rate, latency, and power consumption of modern networks and are thus particularly important for URLLC networks. They provide greater deployment “agility” in deployment, very high capacities thanks to high-frequency communications and reliability through the possibility of multi-connectivity of the “mesh” topology. The solution that we proposed has also the advantage of a relatively low implementation cost, because it is based on a radio architecture that is simpler and cheaper than others. In spite of these advantages, the performance of mmWave radio-frequency systems is severely degraded by a typically strong oscillator phase noise. Therefore, we investigated the use of MIMO systems with energy detection receivers to achieve high-rate communications robust to phase noise. First, the design of the receiver detection algorithm was addressed. In particular, we proposed to use an original and efficient detector based on neural networks. The communication performance was assessed through numerical simulations for uncoded and coded systems. Our results, further detailed in [BDF20], [BFD+22], demonstrated that spatial multiplexing with non-coherent mmWave transceivers can be realized on strongly correlated line-of-sight channels using the proposed detection schemes. Thereby, we highlighted that high-rate mmWave RF systems can be implemented with low-complexity and low-power architectures using MIMO systems with energy detection receivers.

Stochastic geometry framework for ultra-reliable cooperative communications with random blockages (D2.3, Section 3.3). We further focused on cooperative reliable communications for CPSs in industrial environments. Namely, we proposed a solution for an industry automation scenario where a central controller broadcasts critical messages to the wireless devices (e.g., sensors/actuators). We devised a stochastic geometry framework where the rate coverage probability of devices was modelled by taking into account the density of roaming blockages over the factory floor. To alleviate the loss in the coverage, we adopted a two-phase transmission policy, where in the broadcast phase, the central controller broadcasts the messages intended for the devices in the network area. The devices in coverage during the broadcast phase act as decode-and-forward relays in the relay phase, so as to reinforce the signal strength at the devices in outage. The total downlink transmission time is therefore partitioned into two phases by a tuneable factor. Finally, we studied the optimal value of the partitioning factor with varying device densities, blockage densities, and file sizes and we highlighted that a longer transmission time should be allotted to the broadcast phase in the case of larger file sizes or lower transmit power of the controller.

Discontinuous mobile edge computing (D2.3, Section 3.4). A specific contribution in D2.3 was dedicated to scenarios where CPSs are assisted by an edge computing server that can rapidly and efficiently run computational tasks on their behalf. In particular, we described a novel strategy for energy-efficient dynamic computation offloading aimed at minimizing the energy consumption of the overall system, comprising user devices and network elements, under URLLC constraints on the end-to-end service delay and the packet error rate over the wireless interface. To reduce the energy consumption, we exploited low-power sleep operation modes for the user devices, the access point and the edge server, shifting the edge computing paradigm from an always on to an always available architecture, capable of guaranteeing an on-demand target service quality with the minimum energy consumption. To this aim, we introduced Discontinuous Mobile Edge Computing (D-MEC): an online algorithm that dynamically and optimally orchestrates the sleep mode operations and the duty cycles of the network’s elements. In such a dynamic framework, end-to-end delay constraints translate into constraints on overall queueing delays, including both the communication and the computation phases of the offloading service. D-MEC hinges on stochastic Lyapunov optimization, does not require any prior knowledge on the statistics of the offloading traffic or the radio channels, and satisfies the long-term performance constraints imposed by users. Numerical results illustrated the advantages of the proposed method.

End-to-end time synchronization in TSN-5G network architectures (D2.3, Section 3.5). Moreover, we discussed Time Sensitive Networking (TSN), which is a set of standards that provide deterministic communication to standard Ethernet and are being nowadays extended to wireless. An important objective of TSN is to guarantee data delivery with deterministic delay and jitter for real-time applications, which can be a crucial feature for URLLC service and CPSs. We recalled the main features of TSN and briefly described the testbed of Ethernet TSN implemented by CEA, whose evaluated performance on time synchronization was reported in [TBB20].

Channel estimation for non-static users via base station cooperation (D2.3, Section 3.6). Finally, we presented our initial ideas for a novel channel estimation technique in scenarios with non-static users, leveraging cooperation between network access points. These ideas have been further developed into a concrete solution during the second half of the project, which is summarized below, and reported in Section 3.2 of this deliverable.

2.2. Summary of the activities carried out during the second half of the project

We summarize below the work done during the second half of the project, and further detailed in Section 3 of this deliverable.

Design and analysis of MIMO systems using energy detectors for mmWave applications. In Section 3.1, we present the continuation of the work carried out during the first half of the project on mmWave MIMO xHauling. In the context of the activity of Task 2.3, CEA and VSORA collaborated to leverage VSORA's cutting-edge development tools and expertise to implement the proposed neural network based MIMO detector in a real DSP target. While the details of the implementation and the evaluation results are reported in the two deliverables of Task 2.3, we present here the deployment constraints and the development flow. We show that we can quantize the weights of the neural network to only 6- bits with negligible degradation on the performance. Besides, we expect achieving high throughput (5Gbps), with a peak power consumption of only 0.58W. This is equivalent to 0.11nJ/bit. Thus, the proposed quantization scheme and DSP design allow to achieve high throughput and high energy efficiency for mmWave MIMO xHauling.

Joint estimation of CFOs and Doppler shifts in mmWave distributed MIMO systems. In Section 3.2, In Section 3.2, we introduce a Doppler shift estimation technique in a distributed MIMO system. Because of the multiplication of access points and the diverse clock offsets it induces, the synchronization procedure is harder than in conventional MIMO. We show here that one can use the multi transmission to enhance the joint Doppler shift and oscillator offset estimation. It thus limits the effect of channel aging and also reduces the amount of channel feedback.

Transferable and distributed user association policies for 5G and beyond networks. In Section 3.3, we study the problem of user association, namely finding the optimal assignment of user equipment to base stations to achieve a targeted network performance. More specifically, we focus on the problem of *knowledge transferability* of user association policies. Indeed, traditional non-trivial user association schemes are often scenario-specific or deployment-specific and require a policy re-design or re-learning when the number or the position of the users change including change in network topology. In contrast, transferability allows applying a single user association policy, devised for a specific scenario, to other distinct user deployments, without needing a substantial re-learning or re-designing phase. Therefore, such a key feature is particularly important for CPSs as it considerably reduces the computational and management complexity as well as operation costs. To achieve transferability, we first cast user association as a multi-agent reinforcement-learning problem. Then, based on a *neural attention mechanism* that we specifically conceived for this context, we propose a novel distributed policy network architecture, which is transferable among users with *zero-shot generalization capability* i.e., without requiring additional training. Numerical results show the effectiveness of our solution in terms of overall network communication rate, outperforming centralized benchmarks even when the number of users doubles with respect to the initial training point.

Evaluating handover performance for end-to-end LTE networks with OpenAirInterface. In Section 3.4, we present an experimental testbed of the handover procedure using an accessible and reconfigurable software defined radio environment with an end-to-end architecture (i.e. including both the radio access network and the core network). First, we provide a comprehensive overview of the X2 handover procedure, in an end-to-end cellular network architecture and detail the handover condition, message flow and latency decomposition. We then describe our OpenAirInterface based implementation and end-to-end experimentation of LTE X2 handover in a full-SDR environment. Finally, we analyse the performance in terms of end-to-end throughput and of latency for each step of the handover procedure and compare it to the state of the art of X2 handover experimentation.

Resource scheduling optimization for ultra-reliable low-latency communications. In Section 3.5, we propose a dynamic and efficient resource scheduling based on Lyapunov's optimization for Ultra-Reliable Low Latency Communications, taking into account the traffic arrival at the network layer, the queue behaviors at the data link layer and the risk that the applied decision might result in packet losses. The trade-off between the resource efficiency, latency and reliability is achieved by the timing and intensity of decisions and is adapted to dynamic scenarios (e.g., random bursty traffic, time-varying channel). Our queue-aware and channel-aware solution is validated by an experimental testbed using OpenAirInterface and performance is evaluated in terms of latency, reliability and resource efficiency and

Error structure aware parallel BP-RNN decoders for short LDPC codes. In Section 3.6, we address the problem of decoding of short block length Low Density Parity Check (LDPC) codes, relevant to machine-to-machine communications and URLLC scenarios. It has already been demonstrated that Belief Propagation (BP) can be adjusted to the short coding length, thanks to its modeling by a Recurrent Neural Network (BP-RNN). To strengthen this adaptation, we introduce a new training method for the BP-RNN. Its aim is to specialize the BP-RNN on error events sharing the same structural properties. This approach is then associated with a new decoder composed of several parallel specialized BP-RNN decoders, each trained on correcting a different type of error events. Our results show that the proposed specialized BP-RNNs working in parallel effectively enhance the decoding capacity for short block length LDPC codes, improving the reliability of the communication link.

2.3. List of Publications and Patents

We provide below the list of publications and patents resulting from the activities carried out in Task 2.2.

Publications

J. Rosseel, V. Mannoni, V. Savin, and I. Fijalkow, "Error structure aware parallel BP-RNN decoders for short LDPC codes," in *IEEE International Symposium on Topics in Coding (ISTC)*, August 2021.

M. Sana, N. di Pietro and E. Calvanese Strinati, "Transferable and Distributed User Association Policies for 5G and Beyond Networks", *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 966-971, September 2021.

R. Bertolini and M. Maman, "Evaluating Handover Performance for End-to-End LTE Networks with OpenAirInterface", *IEEE 94th Vehicular Technology Conference (VTC Fall 2021)*, October 2021.

G. Ghatak, S. R. Khosravirad , A. De Domenico, "Stochastic Geometry Framework for Ultrareliable Cooperative Communications With Random Blockages", *IEEE Internet of Things Journal*, vol. 9, no. 7, pp. 5150-5161, April 2022.

S. Bicaïs, A. Falempin, J. -B. Doré and V. Savin, "Design and Analysis of MIMO Systems Using Energy Detectors for Sub-THz Applications", *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 3678-3690, June 2022.

M. Merluzzi, N. Di Pietro, P. Di Lorenzo, E. Calvanese Strinati, S. Barbarossa, "Discontinuous Computation Offloading for Energy-Efficient Mobile Edge Computing", *IEEE Transactions on Green Communications and Networking, vol. 6, no. 2, pp. 1242-1257*, June 2022.

S. Bicaïs, A. Falempin, Doré J.-B., and V. Savin, "Design and analysis of MIMO systems using energy detectors for sub-THz applications", *IEEE Transactions on Wireless Communications*, vol. 21, no. 6, pp. 3678-3690, June 2022.

L. N. Dinh, R. Bertolini and M. Maman, "Dynamic Resource Scheduling Optimization for Ultra-Reliable Low Latency Communications: From Simulation to Experimentation", *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, September 2022.

A. Falempin, J.-B. Doré, T. D. Nguyen, and J. Schmitt, "Quantization of Neural Network Demapper for sub-Thz communications" IEEE Vehicular Technology Conference (VTC2022-Fall), September 2022, submitted.

Patents

S. Bicaïs, J.-B. Doré, A. Falempin, “Méthode de démodulation par apprentissage automatique pour récepteurs MIMO à détection d’énergie”, Patent Application No. FR2005670, May 29, 2020.

D. Demmer, “Compensation de dérives fréquentielles dans un réseau multi-antennaires distribué”, Patent Application No. FR2105275, May 20, 2021.

M. Sana, N. di Pietro, E. Calvanese Strinati, and B. Miscopein, “Method of association of user equipment in a cellular network according to a transferable association policy”, U.S. Patent Application No. 17/449,337, September 29, 2021.

3. INNOVATIVE SOLUTIONS FOR CPS COMMUNICATION MODULES AND NETWORKING

3.1. Design and analysis of MIMO systems using energy detectors for mmWave applications

3.1.1. Motivation and related work

Multi-connectivity schemes (based on frequency, time, or spatial diversity) are a way to increase the reliability of a wireless transmission. This feature is critical for CPS application scenarios. From the infrastructure side, reliability can be supported via the deployment of a mesh network system. This requires though extreme xHaul capabilities (bandwidth, latency requirements). These requirements make fibre solutions desirable, but sometimes complicated by local installation constraints to efficiently deploy networks. The wireless infrastructure is foreseen as a complement to the optical fibre deployment as it offers more agility, shorter installation times, and, in the case of a mesh architecture, strong reliability. It may also provide connectivity to mobile, removable or even flying access points. Finally yet importantly, these solutions will be valuable if and only if their cost efficiency is demonstrated. From a spectrum perspective, this paradigm will require large bandwidth and therefore wireless mmWave links in V-, E-, W-, D-bands, namely 60 GHz to 81 GHz, 90 GHz and 140 GHz, are investigated by 3GPP's standardization groups as a complement to 5G mmWave bands (26-28 and 39 GHz). These facts support the search for novel architectural and hardware solutions for cost-effective and high-performance wireless infrastructures.

To achieve high-data-rate mmWave communications, additional research is required to design efficient and new physical layer algorithms. Traditional techniques cannot be directly transposed to mmWave bands (especially for spectrum above 60GHz) as they do not consider the specific features of RF impairments of mmWave systems. In particular, they suffer from strong phase impairments due to the poor performance of high-frequency oscillators [VPZ13]. State-of-the-art approaches [JDC20] investigate the use of coherent systems together with channel bonding. This type of architecture needs to be further combined with signal processing optimizations to mitigate the effects of phase impairments leading to complex practical implementations. Conversely, we consider non-coherent detection for its inherent robustness to phase noise and simple implementation. As an example, fully integrated 260GHz on-off keying (OOK) transceiver is demonstrated in [PKT+12]. Transceivers based on energy detection (ED) have been extensively studied for systems with a single transmit antenna and multiple receive antennas, see [JDPM16] and references therein. Nevertheless, for non-coherent sub-THz systems, the main challenge is to increase the spectral efficiency while maintaining a low complexity. With regard to this objective, the work in [PW09] is relevant as it shows that MIMO systems with amplitude detection receivers may exploit spatial multiplexing to increase their spectral efficiency. Therefore, we investigate the design of MIMO systems with ED receivers to achieve high rate mmWave communications.

In our work, reported in deliverable [D2.3], the channels are strongly correlated, and moreover, the resulting interference is nonlinear due to the ED at the receiver. The strong and nonlinear interference between channels represents a significant challenge to achieve spatial multiplexing with non-coherent transceivers in sub-THz frequencies. Hence, the contributions of our work are the following. First, we derived the model for MIMO systems using ED receivers. Second, the design of the receiver detection algorithm was investigated. We derived the joint maximum likelihood (ML) detector corresponding to the studied nonlinear MIMO channel using a Gaussian approximation approach. In addition, we proposed an original and efficient detector based on neural networks, which does not need any knowledge or assumption of the channel. We also detailed the differences between state-of-the-art detection methods and the two proposed detectors. Third, the system performance was evaluated through numerical simulations. We introduced a realistic scenario modelling a fixed indoor wireless link in the D band at 145 GHz. Our results showed that low-error rate communications can be achieved with strong spatial interference between channels using the proposed demodulation algorithms. Fourth, we considered the integration of a forward error correcting code (FEC) scheme in order to achieve channel coding gains. With regard to targeted high-rate low-complexity applications, we proposed the use of a Bose, Ray-Chaudhuri and Hocquenghem (BCH) code with a short packet length that can be implemented with a low-latency low-complexity decoder. The results of numerical simulations confirmed that the integration of the FEC scheme leads to significant performance gains in terms of achievable data rate.

To avoid content duplication, we refer to [D2.3], Section 3.2, for the details of the above work, carried out during the first half of the project. We also mention that this work led to a patent application [BDF20], and has been published in [BFD+22]. In the following, we report on the implementation of the proposed neural network MIMO detector on a real target, work that has been carried out during the second half of the project.

3.1.2. Implementation of the proposed neural network MIMO detector on a real target

Numerical results reported in [D2.3] were obtained via Matlab simulations. Matlab is an environment with its own programming language for directly expressing mathematics in arrays and matrices. Its API is user friendly but the drawback is the relatively long processing time: in particular, this application could not be embedded in a real-time system as it is. In the context of the activity of Task 2.3 of WP2, CEA and VSORA collaborated to leverage VSORA's cutting-edge development tools and expertise to implement the neural network detector described above onto a real DSP target. The details of the implementation and the consequent evaluation results are reported in the two deliverables of Task 2.3 ([D2.5] and [D2.6]). Here, we only present the deployment constraints and the development flow.

In general, porting a numerically validated algorithm to a real DSP target introduces new constraints:

- **Real time constraint**

  The application is a stream that must be processed periodically. If the processing time is higher than the sampling period, the system will lose packets. The experimental study will explore this constraint and will determine the processing needs (basically the number of ALUs of the system). VSORA's DSP is scalable: we can increase the number of processing units to fit the requirements.

  Another axis of research is the coding style: the DSP has some features (such as the combination of several instructions) which improve the processing capacity. This is theoretically recognized and optimized by the compiler released with the development kit, but in some cases, optimization patterns are not recognized. A fine study of the embedded code could improve the coding style and finally gain performances.

- **Memory constraint**

  The DSP contains several memories: On one hand, a tightly-coupled memory (TCM) located in the core of the DSP is a memory directly connected to the processing units. This memory is fast but has a low density. On the other hand, D-RAM, used as peripheral, can store large amount of data. This memory is dense (small footprint) but it is slow. The connection between these two types of memory is done via direct memory accesses (DMAs). By targeting a real DSP, we will study the needs for both types of memory, the number of DMAs to be implemented and the limitations introduced by them.

- **Quantization constraint**

  The quantization is the operation which cuts numbers to be stored in memory (precision limitation or saturation). This is done on all computer systems, including software like Matlab: for instance the result of $\cos(\pi/2)$ executed by Matlab is not equal to 0, but it gives a very small number. The IEEE consortium defined a standard (IEEE754) to represent floating point values. The common data type used is "double": in this case data is represented over 64 bits: the exponent coded over 11 bits, the mantissa coded over 52 bits plus the sign.

  The type "double" introduces a very small error of quantization, which can be neglected during the algorithm study phase. However, this quantization is too large: we do not need such precision on a real target and the memory footprint is too large.

  VSORA's DSP encodes data following the IEEE754 norm, but the field (exponent, mantissa) width is a static configuration (decided by the customer before the silicon design). The development tools allows to finely study the quantization impact on the system.

As depicted in Figure 3.1-1, the development kit on VSORA's DSP is organized in platforms and each platform has a dedicated goal. They all share the same source code. This allows to "navigate" between the simulation platforms very easily: if a problem is detected on the last simulation platform (FPGA), we can quickly modify the algorithm if needed.

Figure 3.1-1 – VSORA's development platforms.

The **native** platform describes the system in C++. It is released with a mathematical library (vslib) developed by VSORA: it defines containers (matrix, vector,…) and associated operations (linear operations, multiplication, inversion,…). This library follows as far as possible the coding style of Matlab: the porting of code from the Matlab's environment to VSORA's environment should be easy and quick. This platform is used to develop the algorithm.

The **high-level** platform introduces the DSP: the code developed during the native phase is compiled with special options to be mapped on the target: the binary generated is the same for all lower level platforms (up to silicon). Simulations on the high-level platform generate reports about the processing cycles and the memory needs (rough estimation). This platform is also used to study the quantization since the DSP's model has the exact chosen precision.

The **TLM** (transaction-level modelling) and **RTL** (register-transfer level) platforms introduce more accurate models of DSP: These platforms are useful to develop the final silicon, we will not use them in the context of CPS4EU.

The **FPGA** platform is the final one: it runs application on "true" silicon. Only the core of the DSP is mapped over the FPGA. We use remote FPGA through the Amazon Web Service: this allows to use FPGA while optimizing the costs. This platform is used to validate the whole system and to run long simulations.

The final goal of CPSs is to be embedded in autonomous devices. New features and new approach in communication systems will require to be implemented on efficient DSP. However, this implementation could have an impact on the algorithm itself, since the real time aspect, the memory issues and the quantization effect are hard constraints. VSORA's development flow allows to manage these constraints globally and at early stage of development. Finally, simulations on a remote FPGA authorize long duration patterns. This was investigated in "*The simulation tools and experimental platform"* deliverable [D2.5].

3.1.3. Method and environment

We present here the contribution of VSORA during the implementation of the MIMO system on the VSORA platform. VSORA is the provider of the simulation platform which reflects the results that will be get once the system is implemented on a real target.

The global simulation environment of the MIMO system set up by CEA is shown in the diagram below:

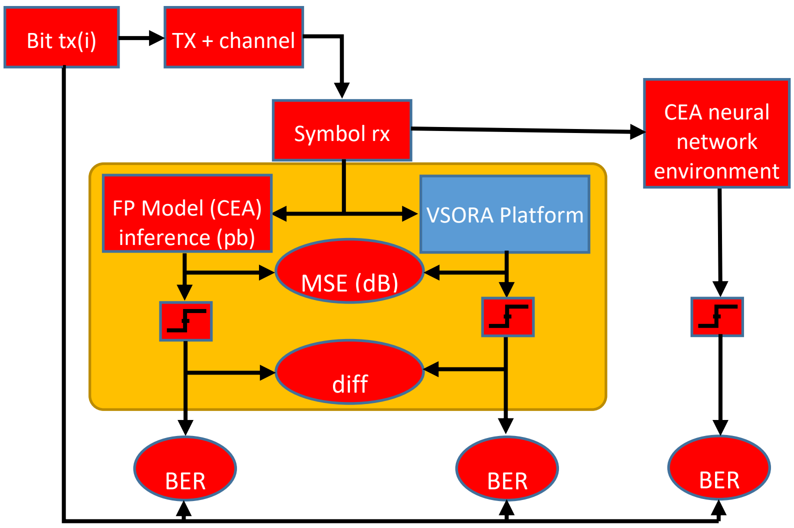


Figure 3.1-2 – *Simulation environment of a MIMO system*

During the project, VSORA provided 8 releases of its simulation platform. As the project from VSORA side was evolving quickly, we decided to regularly propose the most recent version of our DSP to the CEA to conduct their simulations. We took care to make the installation and the update process as simple as possible to facilitate the task of the user.

Release name	Date
vsora-ai_2011_001_cea.tar.gz	2020-11-04
vsora-ai_2104_002_cea.tar.gz	2021-04-16
vsora-ai_2105_000_cea.tar.gz	2021-05-25
vsora-ai_2107_001_cea.tar.gz	2021-07-06
vsora-ai_2108_000_cea.tar.gz	2021-08-14
vsora-ai_2109_001_cea.tar.gz	2021-09-13
vsora-ai_2112_001_cea.tar.gz	2021-12-15
vsora-ai_2202_002_cea.tar.gz	2022-02-22

VSORA also provided the interface environment (yellow box in the diagram) to compare the results of the VSORA platform and the result of the original model of the neural network described with the tensorflow framework. The model of the neural network is exported as pb files. These files can be simulated within the original framework and give a reference for the simulation result.

PB files are also the inputs for the VSORA platform: the model is compiled for the VSORA target (using the compilation tools – graph compiler + LLVM compiler developed by VSORA). The simulations give different results than the DSP model because it processes quantified data. We take advantage of the ability to configure easily the quantification to study its influence on the results.

The simulations have been done with the platform "high level". This platform executes compiled code on a fast environment and gives a good estimation of the CPU load and the exact usage of memory. Data are also quantized and computations are bit to bit equivalent to the real target. At the project definition in 2019, we expected to use the remote FPGA platform to complete the study.

The remote FPGA platform has been developed during the project and it is functional. However, it appeared that it didn't match with the scope of the project. Indeed, the number of base units (number of parallel ALUs) is limited by the capacity of the FPGA and the DSP has a fixed quantification. In order to have a different set of quantification, it is necessary to synthetize a new database. This process is time consuming and the synthesis of many versions of the DSP was out of the scope of the project.

The remote FPGA platform is useful for the algorithmic study: the main use is to speed up long simulations, where one does not care about the configuration (quantification): the process consists only of loading a new algorithm in the memory of the DSP. When the project was submitted, the scope of the study was not precisely defined: originally it was planned to use the classical DSP instructions and we finally switched to the AI instructions since VSORA also started to work on a version of its DSP with this feature. The neural network (algorithm part) used for the MIMO system was already defined and the goal of the joint work between CEA and VSORA was to map this network on a true DSP, and to study the influence of the quantification and of the number of base units. Obviously, the remote FPGA was not suitable with this study.

VSORA helped the CEA in the interpretation of the results and to fine tune the configuration of the system. The DSP is a complex system and it is necessary to know precisely its architecture to understand some phenomena. For example, by increasing the number of ALU of the system, the overall performances may not follow in the same range since the number of MACs (multiplication / accumulation units) varies with the square of the inputs (this is a specific feature of the DSP). In this case it is necessary to increase the batch size (the number of samples handled) in order to avoid idle ALUs.

3.1.4. Results of the study

The study and the related results are exposed in research paper "Quantization of Neural Network Demapper for sub-Thz communications" which was submitted to the VTC2022-Fall (IEEE Vehicular Technology Conference). This paper is a common work between the CEA (Alexis Falempin, Jean-Baptiste Doré) and VSORA (Trung Dung Nguyen, Julien Schmitt). We present here the overall conclusions.

Regarding the impact of the quantized weights on the Bit Error Rate (BER) we compare different sets of quantization with the unquantized model. Unquantized model means that simulations are run using the original data format of the tensorflow framework which is 32 bits floating point; VSORA quantization model is represented using a set of 2 parameters q = (e, m) : the quantization follows the IEEE754 rules with e bits to encode the exponent and m bits to encode the mantissa. The word length is therefore 1 + e + m (adding the sign bit). Vsora floating point supports normalized and denormalized numbers. Only special numbers (NaN, infinity) are ignored since these values have little use in an embedded environment.

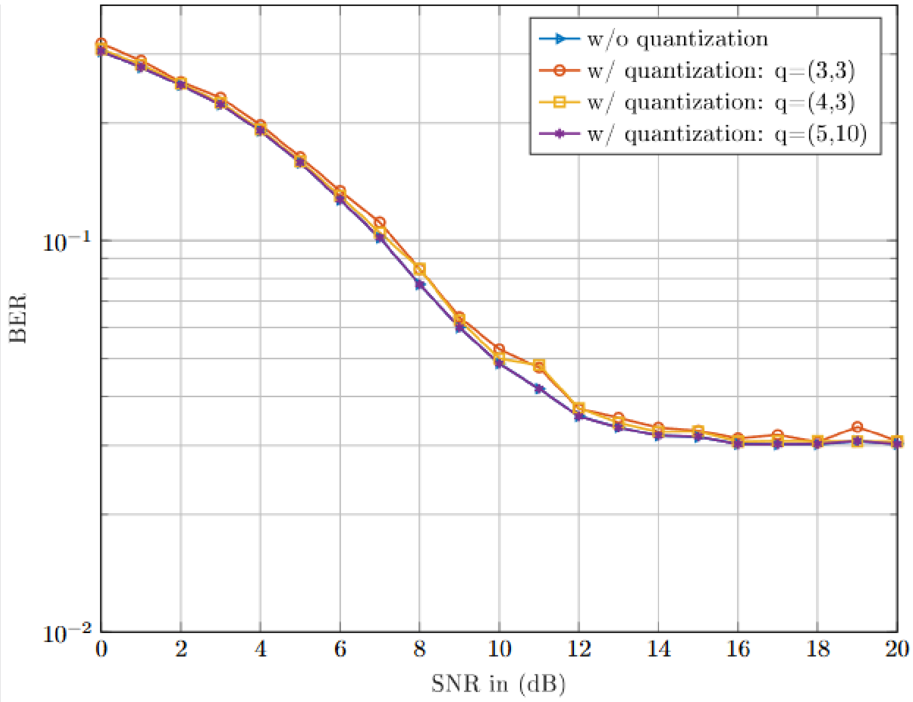


Figure 3.1-3 – *Impact of quantization on BER*

Results presented here are obtained with a post training quantization: this means that we apply the quantization on weights after the training process.

We can observe that using q(5, 10) we obtain the same performances as the unquantified mode. Using q(4, 3) and q(3, 3) will induce a slight degradation which can be neglected. If a real product would be designed, we probably would choose a quantization of q(4, 3) (8-bits words), associated with a QAT (quantization aware training) : the limited word length have a huge impact on the silicon area (mainly in the TCM) and consequently on the price of the final solution.

Another aspect of the study is the throughput obtained regarding the processing power. We assume the clock frequency of the DSP at 2 GHz. We measure the throughput for many configurations of DSP (changing the number of MACs). Each configuration gives a processing power we can express in TFLOPs.

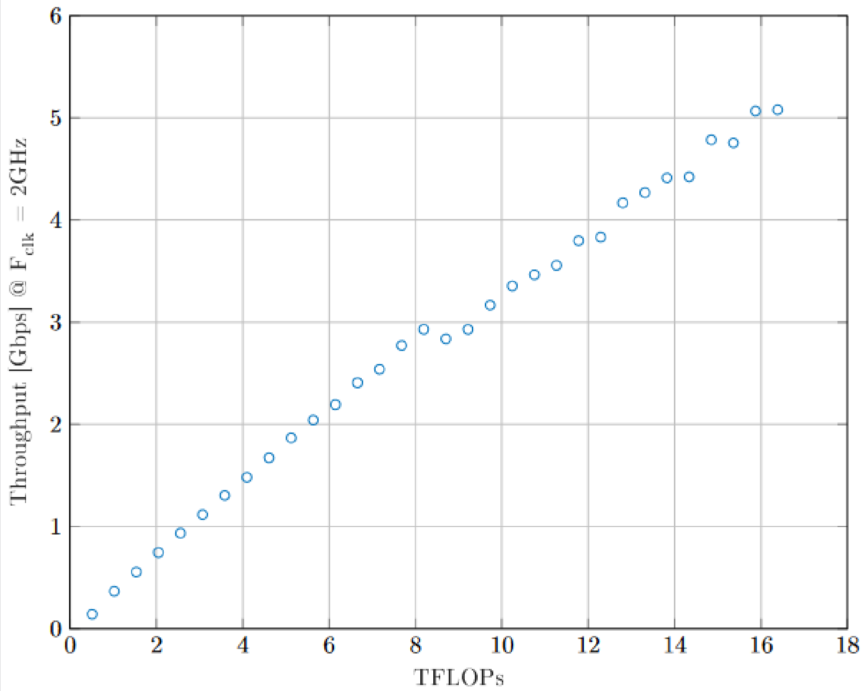


Figure 3.1-4 – *Achievable throughput in Gbps in function of the processing power*

We observe a good linearity between the processing power and the throughput. This linearity has been obtained by modifying the coding style: in particular, it has been necessary to increase the batch size to keep an activity on the maximum computing units. We chose a batch size of 32k symbols to keep the same configuration independently from the number of MACS and which avoid idle processing units and keep an acceptable latency. This exercise highlighted the difficulties of achieving a good performance in an embedded device (compared to a high level code in a standard framework). It also proved the efficiency of the development method proposed by VSORA: even if we had to rework the code to reach the desired performance, we kept a unique code which is still described with a high level of abstraction. Traditional methods require writing a dedicated code for the embedded device, increasing the time of development, with a possible loss in efficiency and reliability.

3.1.5. Conclusion

In conclusion, the implementation of and advanced MIMO system on a real target, using artificial intelligence was successful. At the beginning of the project, the AI core proposed by VSORA was at the very early stage of development: we achieved to develop the core supporting a large variety of neural network topologies. The compilation paradigm imagined by VSORA (separate the framework from the target compilation) has been successfully proven by the CEA: the mapping of the neural network required very few efforts of development. Although the remote FPGA platform was not used by the end-user in the scope of the study, we still achieved its development and now we have a clearer vision of the benefits this platform can contribute. CPS4EU was clearly helpful to VSORA to develop the AI version of its DSP, to refine the release process, to finalize the documentation.

3.2. Joint estimation of CFOs and Doppler shifts in mmWave distributed MIMO systems

3.2.1. Motivation and related work

Wireless connectivity is making its way into industrial networks as it is “plug and play”, eases the maintenance and control of devices, and even enables the coordination of moving devices such as transport vehicles. Sub-GHz bands are not suitable to support a dense traffic and their reliability can be significantly reduced because of the presence of interfering systems. That is why millimeter wave (mmWave) spectrum is considered as a plausible candidate to replace wires in industrial sectors especially with the licensed band at 28 GHz and the unlicensed band at 60 GHz [SOL18]. However, wireless communications are unreliable compared to wired links and poor connectivity can result in delayed transmission or reduced performance. The challenge is thus to ensure Ultra Reliable and Low Latency Communications (URLLC) wirelessly which is critical for industrial applications.

In mmWave communications, Radio Frequency (RF) blockages may occur because of obstacles and moving objects and vehicles. Spatial diversity is therefore required to prevent those RF blockages [KHO19]. In this context, distributed MIMO systems, also known as Cell-Free MIMO, have been proposed to improve the system capacity of conventional co-localized systems [BJO20, DAW06]. With such architectures, the Base Station (BS) is composed of a Centralized Unit (CU) responsible for the digital signal processing and several Radio Units (RUs) that are spatially distributed over the cell and responsible for the RF processing of the signals. In this work, we consider beamforming techniques at the RUs side to come up with severe path loss of the millimeter wireless links. The main advantages of the distributed architecture are i) providing an improved coverage thanks to an enhanced macro diversity and higher robustness to RF blockages and ii) less path loss and energy consumption thanks to shorter BS-device distances in average [NGO17, ZHA20]. The uplink signals are received by one or more RUs which transpose it into the baseband and forward it to the CU for reconstruction. This is known as joint reception. When it comes the downlink, the CU precodes the signal to transmit to the devices and send the resulting signals to the RUs for over-the-air transmission. The devices thus receive a set of coherent signals to recombine for reconstruction. It is known as joint transmission. The way to precode the downlink messages depends on the system. It can be for instance linear schemes like Zero-Forcing or Minimum Mean Square Error to mitigate the inter-user interference in Multi-User MIMO (MU-MIMO) [BJO20a].

The capacity of a MIMO system highly depends on the quality of the Channel State Information (CSI). First, reliable instantaneous CSI is hard to ensure in a MU environment because of possible pilot contamination. Then, channel aging limits the validity of the CSI in time. Channel aging results from inevitable residual Carrier Frequency Offsets (CFOs) and UE motion inducing additional Doppler shifts. This phenomenon is even more critical in distributed MIMO architectures because the wireless links between a UE and the different RUs experience different frequency offsets. Besides in mmWave communications, because of the large frequency offsets (both CFOs and Doppler shifts) the channel aging is a significant impairment.

When it comes to joint-transmission, the UE devices must perform the estimation of multiple frequency offsets in order to reconstruct the downlink signal. Some techniques based on training sequences exist and some of them can even approach Maximum Likelihood (ML) performance [ZAR08]. However, those iterative algorithms are rather computationally intensive and are therefore not suitable for devices with limited computation resources or links with stringent latency constraints like with industrial networks. Based on this observation, we believe that it is not suited to perform the multi-CFO estimation at the UE side. That is why in this study, we move the estimation at the BS side based on measurements of the uplink signals like in [ZEN18]. Because the respective contributions of the CFO and the Doppler shifts are the same on the uplink and downlink [ZEN18], it is thus required to jointly estimate the CFOs and the Doppler shifts for each link. In this work, we thus propose to tackle the challenge of frequency estimation in distributed MIMO architectures. The main contribution of this study is not about the frequency estimation at the RU level (based on pilot or training sequence) but at the BS level (based on multiple frequency offset measurements) in order to separate the respective contributions of the CFO and the Doppler-induced shifts. For our proposed joint frequency estimator, there is no need for a priori knowledge on UE motion and there is no condition on channel delay spread. It is a solid contribution with respect to literature solutions and [ZEN18].

3.2.2. *System model*

The BS is composed of N RUs responsible for the transmission/reception of wireless signals and one CU performing the signal processing. Each RU is equipped with an antenna array and is able to perform a beamforming with beam width α. The K devices are equipped with omnidirectional antenna.

In this study, we assume that the antenna array is able to steer its beam in the direction of the maximum received power, either in Line-of-Sight (LOS) or in non-LOS. The multi-path components of the propagation channel are then filtered out by the RU beamforming [RAP15]. The received signal is thus only composed of one resolvable path along with L_p unresolvable paths. Besides, the RUs are static which implies that the Doppler effect is due to the mobility of the devices. However, the AoDs are equal for all the unresolvable paths between device i and RU j, and therefore the Doppler shift is the same. As a consequence, a simpler propagation channel model with a unique unresolvable path will be assumed for the rest of this work. Nonetheless, the Doppler shift experienced on each device-RU link varies because the AoDs are different.

In addition to the Doppler effect, the frequency shift induced by oscillator imperfections are taken into consideration as well. $f_j^{\{RU\}}$ and $f_i^{\{UE\}}$ respectively denote the CFOs of the RU j and the device i. The system model for two RUs and two devices is depicted in Figure 3.2-1. For the RU, a common clock signal can be obtained with a network synchronization protocol like Global Navigation Satellite Systems (GNSS) or Time Sensitive Networking (TSN), which is becoming more and more central in industrial Internet of Things (IoT). And when the synchronization provided by the protocol is not sufficient, other over-the-air synchronization procedures can be considered [ROG14]. Therefore, the clock imperfection of the RUs will be neglected for this study: $f_j^{\{RU\}} = 0$ for any RU j.

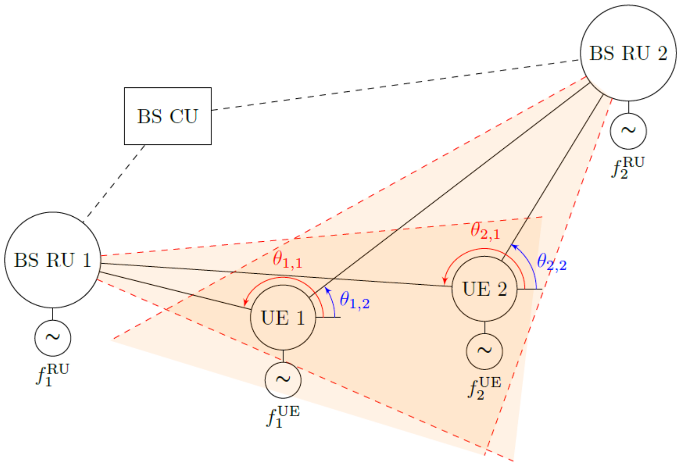


Figure 3.2-1 – System model with two UEs.

The resulting frequency offset experienced on the uplink path between device i and the RU j is thus the sum of the contributions of the device CFO $f_i^{\{UE\}}$ and the Doppler shift $f_{\{i,j\}}^{\{DOP\}}$:

$$\Delta f_{\{i,j\}}^{\{UL\}} = f_i^{\{UE\}} + f_{\{i,j\}}^{\{DOP\}} \tag{3.2-1}$$

The Doppler shift expression is given in (3.2-2) where v_i denotes the device speed, f_s the carrier frequency of the signal, c the light celerity, $\theta_{\{i,j\}}$ the AoD of the link and ϕ_i the device angular direction.

$$f_{\{i,j\}}^{\{DOP\}} = \frac{v_i f_s}{c} cos(\theta_{\{i,j\}} - \phi_i) \tag{3.2-2}$$

When it comes to the frequency offset experienced on the downlink for the same path, one obtains the expression (3.2-3).

$$\Delta f_{\{i,j\}}^{\{DL\}} = -f_i^{\{UE\}} + f_{\{i,j\}}^{\{DOP\}} \tag{3.2-3}$$

The overall frequency offset experienced on the link depends on the same terms but with the opposite contribution for the device CFO. Indeed, as the relative motion of the device with respect to each RU is the same for both links then the Doppler shift is also the same on both UL and DL links. To obtain this expression, it is assumed that the time interval between the uplink and downlink signals is small enough so that the CFO and the device localisation remain constant which is a reasonable assumption.

This observation justifies the need of a joint CFO and Doppler shift estimation on each wireless link. It is indeed required to estimate separately the contributions of $f_i^{\{UE\}}$ and $f_{\{i,j\}}^{\{DOP\}}$ to properly estimate the overall frequency offset. We propose in this study, a joint estimation technique based on measurements from the uplink signals. The resulting estimation of the downlink frequency offset $\Delta f_{\{i,j\}}^{\{DL\}}$ can thus be used to compensate the phase variation of the Channel State Information (CSI) induced by time delay between the CSI and the DL precoding.

One can observe that the estimation problem can be performed independently for each device. The estimation problem introduced in (3.2-1) can thus be simplified to a single-device problem where the index i is omitted:

$$\Delta f_j = f^{\{UE\}} + f^{\{DOPm\}} cos(\theta_j - \phi) \tag{3.2-4}$$

3.2.3. *Joint-Estimation algorithm*

- **Regression Problem**

We aim at estimating the device CFO $f^{\{UE\}}$, the maximum Doppler shift $f^{\{DOPm\}}$ and the angular direction of the device ϕ based on the measurements of the AoD of the wireless uplink signals θ_j at each RU and the cumulated frequency shifts Δf_j.

On the one hand, the quality of the measurement for the frequency shifts depends on many parameters: the signal-to-noise ratio of the wireless link, the reference signals (preamble structure [LI01], pilot-based [SPE99], blind [PAN06] and so on). Given that our objective is to assess the feasibility and the performance in ideal conditions, the overall frequency shift measurement is assumed to be ideal (i.e. the estimation of Δf_j is noiseless).

On the other hand, the measurement of the AoDs cannot be ideal. Indeed, the AoD is measured at each RU thanks to the angle-domain projection of the signal [ZEN18] obtained with the antenna arrays. The measurements of the AoDs, denoted $\hat{\theta}_j$ are then noisy and the accuracy depends of the beam width :

$$\hat{\theta}_j = \theta_j + \epsilon_j$$

where ϵ_j denotes a random noise that we assume follow a uniform continuous distribution $\mathcal{U}(-\alpha/2, \alpha/2)$.

The regression problem described in (3.2-4) is non-linear and therefore not easy to solve. We propose here a pragmatic solution to linearize the problem by using trigonometric relations:

$$\Delta f_j = f^{UE} + f^{DOPm} \cos(\phi) \cos(\hat{\theta}_j) + f^{DOPm} \sin(\phi) \sin(\hat{\theta}_j) \tag{3.2-5}$$

The expression (3.2-5) can be put in the matrix form as well:

$$\mathbf{\Delta f} = \mathbf{X}\hat{\mathbf{Y}} \tag{3.2-6}$$

Where

- $\mathbf{\Delta f} = [\Delta f_1, \dots, \Delta f_N] \in \mathbb{R}^{1\times N}$
- $\mathbf{X} = [f^{UE}; f^{DOPm} \cos(\phi); f^{DOPm} \sin(\phi)] \in \mathbb{R}^{1\times 3}$
- $\hat{\mathbf{Y}} \in \mathbb{R}^{3\times N}$ with j^{th} column $\hat{Y}_j = [1; \cos(\hat{\theta}_j); \sin(\hat{\theta}_j)]^T$

<u>**Lemma 1:**</u> Assume that $N \geq 3$, and AoDs $\theta_1, \dots, \theta_N$ are independent and uniformly distributed in $[0, 2\pi]$. Then, $\hat{\mathbf{Y}}$ is full rank with probability 1.

As a consequence, when $N \geq 3$ $\hat{\mathbf{Y}}\hat{\mathbf{Y}}^{\mathbf{T}}$ is invertible with probability 1 and $\hat{\mathbf{X}}$ can be evaluated by applying (3.2-7):

$$\hat{\mathbf{X}} = \Delta\mathbf{f}\hat{\mathbf{Y}}^T(\hat{\mathbf{Y}}\hat{\mathbf{Y}}^T)^{-1} \tag{3.2-7}$$

Finally, the estimation of the three parameters of interests can be expressed as follows:

- $$\hat{f}^{UE} = \hat{X}(1) \tag{3.2-8}$$
- $$\hat{f}^{DOPm} = \sqrt{\hat{X}(2)^2 + \hat{X}(3)^2} \tag{3.2-9}$$
- $$\hat{\phi} = \text{atan}(\frac{\hat{X}(3)}{\hat{X}(2)}) \tag{3.2-10}$$

where $X(i)$ is the i-th coordinate of the vector $\mathbf{X}$.

➢ Analytical Analysis

The theoretical performance of the estimator is discussed in this section.

By inserting (3.2-6) into (3.2-7), one obtains:

$$\hat{\mathbf{X}} = \mathbf{X}\mathbf{Y}\hat{\mathbf{Y}}^{\mathbf{T}}(\hat{\mathbf{Y}}\hat{\mathbf{Y}}^{\mathbf{T}})^{-1} = \mathbf{X}(\frac{1}{N}\mathbf{Y}\hat{\mathbf{Y}}^T)(\frac{1}{N}\hat{\mathbf{Y}}\hat{\mathbf{Y}}^T)^{-1}$$

Where $\mathbf{Y} \in \mathbb{R}^{3\times N}$ with the j-th column $Y_j = [1; \cos(\theta_j); \sin(\theta_j)]^T$.

__Lemma 2:__ Assume that AoDs $\theta_1, \dots, \theta_N$ are independent and uniformly distributed in $[0, 2\pi]$. Then, the following equalities hold with probability 1.

1. $$\lim_{N\to\infty} \frac{1}{N}\hat{\mathbf{Y}}\hat{\mathbf{Y}}^{\mathbf{T}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix}$$
2. $$lim_{N\to\infty} \frac{1}{N}\mathbf{Y}\hat{\mathbf{Y}}^{\mathbf{T}} = \frac{1}{N}\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{\alpha}\sin\left(\frac{\alpha}{2}\right) & 0 \\ 0 & 0 & \frac{1}{\alpha}\sin\left(\frac{\alpha}{2}\right) \end{pmatrix}$$

By using the above lemma, one gets:

$$\lim_{N\to\infty} \hat{\mathbf{X}} = \mathbf{X}\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{2}{\alpha}\sin\left(\frac{\alpha}{2}\right) & 0 \\ 0 & 0 & \frac{2}{\alpha}\sin\left(\frac{\alpha}{2}\right) \end{pmatrix}$$

It implies that

- $$lim_{N\to\infty}\hat{f}^{UE} = f^{UE} \tag{3.2-11}$$
- $$\lim_{N\to\infty} \hat{f}^{DOPm} = \frac{2}{\alpha}\sin\left(\frac{\alpha}{2}\right) f^{DOPm} \tag{3.2-12}$$
- $$lim_{N\to\infty}\hat{\phi} = \phi \tag{3.2-13}$$

It means that the estimators (3.2-8) and (3.2-10) are consistent but (3.2-9) is not. However, by setting (3.2-14) one gets (3.2-15).

$$\hat{\hat{f}}^{DOPm} = \frac{\alpha}{2\sin\left(\frac{\alpha}{2}\right)} \hat{f}^{DOPm} \tag{3.2-14}$$

$$\lim_{N\to\infty} \hat{\hat{f}}^{DOPm} = f^{DOPm} \tag{3.2-15}$$

We have then defined three consistent estimators for the parameters of interest (3.2-8), (3.2-10) and (3.2-14). It means that by considering the system model presented in (3.2-4) when one collects a large number of measurements of frequency offsets, the three estimators converge in probability to the true value of the parameters independently of the value of the RU beam width α.

3.2.4. Performance Evaluation

In this section, the algorithm performance is evaluated for practical implementations (i.e. with limited number of RUs). Three criteria will be assessed: i) the estimation accuracy of the parameters of interest, ii) the quality of the resulting channel prediction and iii) the achievable sum-rate.

Table I
SIMULATION PARAMETERS

Signal	
Center frequency	26.7 GHz
Channel model	geometrical approximation based on ray-tracing
Propagation Channel	LOS only
CFO	0.1 ppm [15]
RUs	
Location	Uniformly distributed on a 50m radius disc
UEs	
Location	Uniformly distributed
Speed	Uniformly distributed between 0 and 30 km/h
Direction	Uniformly distributed between 0 and 2π

An mixed static/moving UE scenario is considered for the simulations with wireless connectivity working at the 26.7 GHz licensed band [SOL18] The channel model and the list of the simulation parameters is given in Table I. The algorithm defined in [ZEN18] (where UE motion is known) will be used as reference. The comparison with [ZEN18] allows us to evaluate the *price to pay* to estimate the UE direction in addition to the frequency offsets. For the evaluation performance, the normalized Mean Square Error (nMSE) indicator is used in this study. Its expression is reminded in (3.2-16) where Z and $\hat{Z}$ are respectively the real and the estimated scalar values of the parameter of interest, |z| depicts the absolute value of the real z and $\mathbb{E}$ is the expectation operator.

$$\mathrm{nMSE}\left(\hat{Z}\right) = \mathbb{E}\left(\frac{|\hat{Z} - Z|^2}{|Z|^2}\right) \tag{3.2-16}$$

The first performance indicator is the quality of the estimation technique for the three parameters of interest: the device CFO $f^{\{UE\}}$, the device max Doppler $f^{\{DOPm\}}$ and the device angular direction ϕ (not for [ZEN18] because it is known a priori). The results are depicted on Figure 3.2-2.

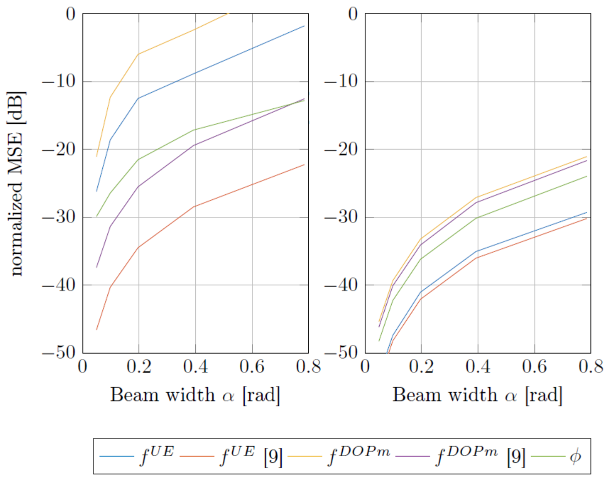


Figure 3.2-2 – Evaluation of the estimation accuracy for two possible RU deployments N=4 (left) and N= 8 (right) ([9] is [ZEN18]).

First, one can estimate the relative impact of the numbers of RUs and the beam width on the estimator accuracy. Indeed, with a large number of RUs (like 8 (right)), the constraints on the beam width to satisfy are less tight. It means that even with large beams, the estimator can provided accurate results. On the contrary, for a reduced number of RUs (like 4 (left)), the beam width must be narrow enough to provide satisfactory accuracy. The latter case is more interesting because it corresponds to a more practical scenario.

It seems worth noticing that the estimators appears to be also consistent when α tends to 0 for any $N \geq 3$. And, for realistic values of α and N, α looks to play a more predominant role on the estimator accuracy. This implies that the quality of the estimation is better improved by narrowing the beam widths than by adding new RUs.

One can observe that the estimation accuracy of the device direction ϕ mainly depends on the number of RUs. Indeed for a limited number of RUs (like 4), it is the parameter know with the poorest accuracy. One can therefore expect larger performance penalty with the algorithm defined in [ZEN18] with limited number of RUs. To assess the performance difference between the two estimators, we propose to evaluate the accuracy of the overall frequency shift at the RU side (including CFO and Doppler shift). The results are depicted in Figure 3.2-3 for 4 and 8 RUs.

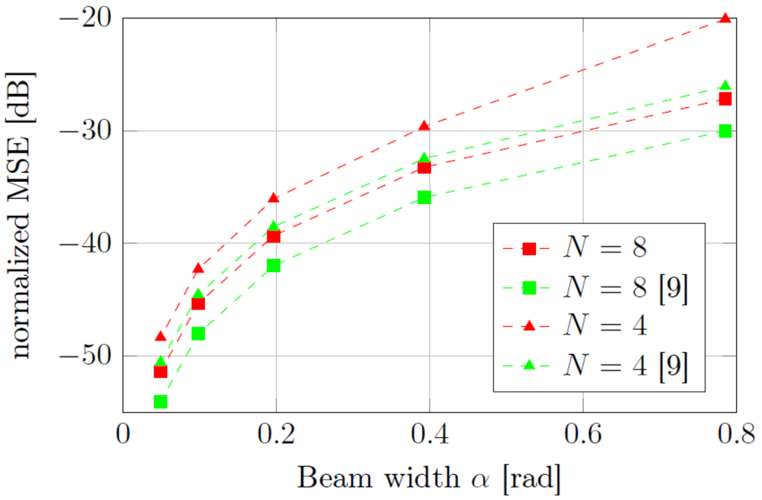


Figure 3.2-3 – Evaluation of the estimation accuracy for the overall frequency shift for two possible RU deployments ([9] is [ZEN18]).

First of all, the proposed solution (with estimation of device direction ϕ) provides good estimation of the overall frequency shifts. As foreseen, the performance penalty with respect to [ZEN18] is larger for a limited number of

RUs. However, it seems interesting to notice that the performance penalty lowers when the beam width α is reduced. In other words, for narrow beam widths, the *price to pay* in terms of estimator accuracy to estimate the device direction as well is negligible.

We can also evaluate the benefits of the proposed method by looking at the resulting average downlink sum-rate per user as a function of the time delay between the CSI acquisition and the MU downlink precoding computation. The objective of this KPI is to observe the effects of channel aging on downlink performance and to evaluat the robustness of the proposed estimation technique against this effect. The results are depicted in Figure 3.2-4. The expression of the mean sum-rate $\mathcal{R}$ is given in (3.2-17) where P_{sig} , P_{noi} and P_{int} respectively denote the powers of the received useful signal, the noise and the interference terms. The noise power if fixed to ensure a Signal-to-Noise Ratio (SNR) of 20 dB per link, i.e. $\frac{P_{\text{sig}}}{P_{\text{noi}}} = 20$ dB. The interference term comes from the inter-user interference from the zero-forcing precoding induced by the error of CSI due to imperfect channel prediction. The curves corresponding to the `ideal' case, meaning perfect estimation and channel prediction and `no compensation' case, meaning no estimation and channel prediction is performed, are plotted as well for comparison.

$$\mathcal{R}(t_1 - t_0) = \mathbb{E}\left(\log\left(1 + \frac{P_{\text{sig}}}{P_{\text{int}}(t_1 - t_0) + P_{\text{noi}}}\right)\right) \tag{3.2-17}$$

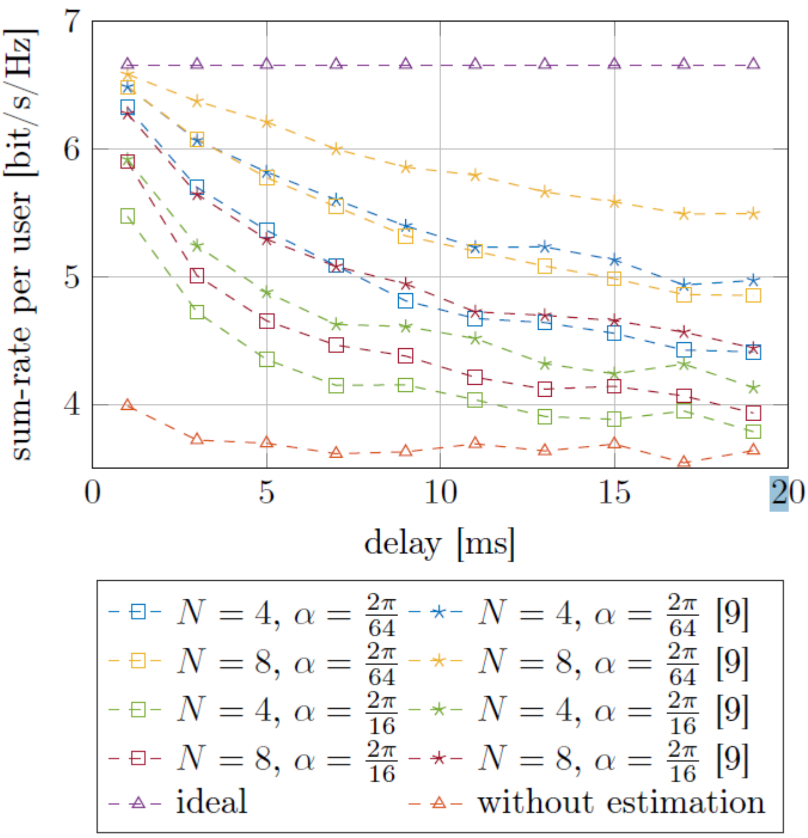


Figure 3.2-4 – Achievable sum-rate ([9] is [ZEN18]).

One can first observe that thanks to the joint CFO and Doppler estimation, proposed in our work and [ZEN18], the performance penalty induced by channel aging is reduced. Indeed, the mean sum-rate is improved with respect to the `without estimation case` which means that the CSI has a better validity in time which improves the downlink capacity. As already stated, the performance provided by the proposed solution is slightly below to what provides [ZEN18]. It is expected and is a direct consequence of what we observed in Figure 3.2-2 and Figure 3.2-3. Nonetheless, the gain in performance is still substantial for a delay of 10 ms, typical frame duration in 5G systems, with 127% increase for $\alpha = 2\pi/64$ and N=4 or 140% increase for $\alpha = 2\pi/64$ and N=8. It shows that when there is no a priori knowledge on UE location and motion, the proposed joint frequency offset estimation technique proposed in our work efficiently improve the quality of the CSI and limits the effect of channel aging. That is a major contribution with respect to state-of-the-art techniques.

Another observation worth making is that it is more interesting to narrow the beam width than to add extra RUs to improve the algorithm performance. It makes the proposed solution more scalable and easy to implement for concrete and realistic distributed MIMO systems.

3.2.5. *Conclusion*

We proposed a joint estimation technique to cope with the frequency offsets in a multi-user distributed MIMO systems. The proposed algorithm works without any a priori knowledge of user motion and the estimation is performed at the BS side to keep the device processing as simple as possible. It is a real advantage with respect to similar solutions proposed in the literature. The proposed estimation technique has suitable asymptotic properties and provides significant performance gains with both realistic and practical scenarios. It makes the proposed solutions appealing for beyond 5G MU-MIMO scenarios.

In our work, strong assumptions have been made (ideal frequency estimates and perfect RU oscillator synchronization). Those assumptions lead to a simple system model which is interesting for a first approach but their impacts on the achievable performance will be analyzed and evaluated in a future publication to consider realistic system constraints.

3.3. Transferable and Distributed User Association Policies for 5G and Beyond Networks

3.3.1. Motivation and related work

With the significant proliferation of smart connected devices, the cyber and physical spaces are fusing, turning humans, objects, and events more and more into sources of digital information. As a result, mobile data traffic is growing exponentially and we are witnessing the emergence of new performance-demanding services such as online gaming, e-health, augmented and virtual reality applications. Some of the challenges induced by these services are being addressed by current fifth-generation (5G) wireless communication systems, for example through the adoption of millimeter-wave (mmWave) technologies as well as massive MIMO and network densification. However, many technical issues, for instance related to high-precision manufacturing in Industry 4.0 or holographic communications, still remain unsolved or need more efficient solutions in terms of efficiency, complexity or deployability, pushing towards sixth-generation (6G) networks [ECS19]. To address these issues, among the various solutions under consideration, the adoption of artificial intelligence (AI) at the network edge (edge intelligence) is envisioned. In this scenario, multiple distributed AI-powered devices can learn and possibly share their knowledge with each other to optimize some network utility functions and achieve some common goals [ECS19], [EPEL20]. This is currently enabled by endowing mobile devices with AI algorithm computing capabilities [YDG19]. In this context, we are interested in adopting edge intelligence to solve the problem of efficient association between users and network access points. Indeed, user association plays a crucial role in mobile communications as it directly affects the network spectral efficiency and the users' perceived quality of service. Its optimization is a difficult task, since it generally involves the solution of non-convex and combinatorial problems with exponential complexity in the number of user devices in the network. Moreover, multiple factors impact the user association and further make difficult its management:

i) the wireless channel dynamics,

ii) the network data traffic dynamics, and

iii) the users' mobility.

This difficulty is even exacerbated in mmWave networks due to severe path loss and high sensitivity to blockage [ADD17], and in large-scale networks due to dense deployment of user equipments (UEs) and base stations (BSs).

Current state-of-the-art solutions of user association are in general not scalable and tangibly lack adaptability. In particular, such approaches, are often grounded on quite rigid assumptions, such as pre-sized and fixed sets of BSs and static UEs, favourable channel conditions, absence of inter-cell or intra-cell interference, full-buffer network traffic. Yet, in dynamic mmWave networks, especially in dense networks, the number of UEs, their position with respect to each other and BSs, and the performance requirements of the services they access, are likely to change over time and are characterized by a high dynamicity. Even in relatively stable scenarios from the radio channel and data traffic points of view, the arrival in the network or the departure from the network of one or more users has an impact on the overall network performance, which requires a constant adaptation of the user association to dynamically guarantee the best possible quality of service (QoS). To tackle this problem of adaptability and scalability, we propose transferable user association policies. Transferability is indeed an important key feature. It allows transferring the user association knowledge acquired in one specific scenario to another one [SJP10], thus, resulting in a significant gain in terms of signalling and computation overhead. With this feature, a UE that attempts to access a cell can benefit from the already available knowledge of other UEs in the network to select the appropriate BS, without requiring any additional training procedure of the global policy. As a direct result, the proposed mechanism adapts well by design to variations in the number of UEs or changes in the geometry of the network, i.e., in their geographical positions, and or to the channels and traffic dynamic.

The user association problem has received a wide attention from the literature. Numerous solutions have been proposed utilizing different techniques to optimize the network performance with respect to (w.r.t.) various metrics [DL16]. In [GA15], the authors neglect the interference to derive a distributed user association to maximize the network sum-rate. A load balancing user association is proposed in [AA19a], with a polynomial-time algorithm to balance the radio resources across BSs. Leveraging a game-theoretical approach, the user association is reformulated as a non-cooperative game with local interactions in [YL18] and as a matching game in [AA19b]. These works either fail to consider environment dynamics such as interference, channels and UEs' traffic dynamics or are subject to a large amount of signalling. Most recent works on user association involve machine learning and reinforcement learning to cope with user association complexity and the radio environment dynamics [QM18], [LB08]. In this sense, a deep neural network (NN) architecture is introduced in [PZ19] that predicts the user association and power allocation. Similarly, authors in [RL20] have formulated the problem of user association with multi-connectivity as a multi-label classification problem. All these works are based on building cumbersome databases, requiring extensive computation, hence resulting unsuitable to large-scale scenarios. To address this issue, a distributed user association scheme based on multi-agent reinforcement learning is proposed in [NZ18]. Although this solution addresses the issue of the database size, it still lacks scalability as their proposed NN architecture intrinsically scales with the number of UEs in the network. In addition, the solutions of the state-of-the-art either optimize the user association for a fixed number and position of UEs in the network or devise user association policies, which are not transferable, i.e., the knowledge of already present UEs in the network cannot be straightforwardly and efficiently transferred to newly arrived UEs.

In this work, we address the problem of transferability of the user association policy. We propose a novel policy network architecture (PNA) and learning mechanism to derive a transferable user association strategy able to address changes in radio environment, including channel dynamics, mobility of UEs as well as the variability of the number of UEs over time.

In summary, the contribution of this work is four-fold:

- Transferability: in contrast to previous approaches from the literature [NZ18], [PZ19], which require reconstructing the PNA (i.e., the NN architecture) and an entire new learning process whenever the number of UEs varies, our new proposed solution presents the advantage of being transferable. In other words, both the PNA and the learned association skills can be transferred to a newly arrived UE that joins the coverage area, without any additional changes. To achieve this, instead of having one specific policy per UE as in [NZ18], we construct a unique global PNA using the mechanism of attention NNs, which can be efficiently trained with the experiences of all UEs. Simulation results show that empowering our proposed architecture with an attention mechanism enables transferability without any additional loss in performance.
- Hysteretic proximal policy optimization: we optimize the user association policy using a policy gradient algorithm, in particular, the proximal policy optimization (PPO) framework. However, dynamic channels and network traffic combined with the simultaneous interaction of agents, make the radio environment strongly nonstationary, which challenges MARL systems. Hence, to stabilize the learning process and improve the convergence, we leverage the concept of the hysteretic Q-learning [LM07], [SO17] to modify the PPO algorithm by introducing two clipping factors that induce a hysteretic behavior in policy updates. We show through numerical simulations the benefit of such a method both on the convergence and the system performance.
- Zero-shot generalization: in addition and in sharp contrast with existing solutions, the proposed mechanism has zero-shot learning capability, i.e., it can actively adapt to the variations due to the departure or arrival of UEs without requiring additional training iterations. For this purpose, we introduce a UE dropout mechanism, which consists in masking some UEs during the learning process to enable robustness of the learned policy w.r.t. the variation of the number of UEs in the network. We show that the dropout mechanism further stabilizes the learning process and enables better knowledge transferability. Also, we show that with this zero-shot generalization capability, our algorithm can quickly adapt to scenarios with mobility.

3.3.2. System model

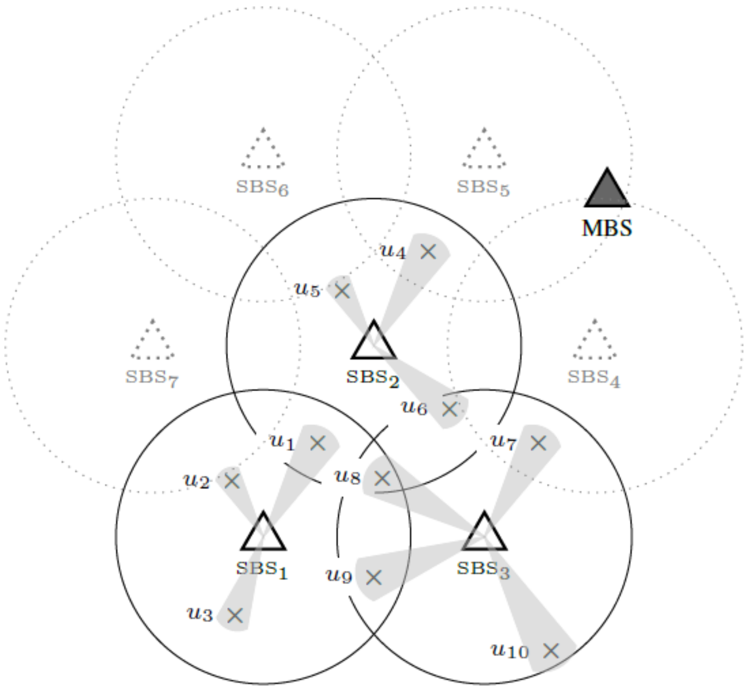


Figure 3.3-1 – Network topology for 3 SBSs, 1 MBS, and 10 UEs.

Network Model. We consider the system model of Figure 3.3-1, focusing on downlink communications. We consider that $K(t)$ UEs are located at time t in a region of the bi-dimensional Euclidean space, covered by N_s mmWave small base stations (SBSs) and a sub-6 GHz macro base station (MBS), to enable ubiquitous network coverage. We denote by $\mathcal{A} = \{0, 1, \dots, N_s\}$ the set of BSs, where 0 indexes the MBS, and by $\mathcal{U} = \{1, 2, \dots, K(t)\}$ the set of UEs in the network. We call a network deployment $\mathcal{D}_k$, a collection of positions of all UEs in the network:

$$\mathcal{D}_k(\mathrm{t}) = \left\{\left(x_j^{(k)}(t), y_j^{(k)}(t)\right), j \in \mathcal{U}\right\},$$

where $x_j^{(k)}$ and $y_j^{(k)}$ denote the coordinates of UE j in deployment $\mathcal{D}_k$ expressed w.r.t. a reference system common to all UEs and BSs. We denote $\mathcal{A}_j \subseteq \mathcal{A}$ the subset of BSs that can be associated with UE j. In general, $\mathcal{A}_j$ is strictly contained in $\mathcal{A}$, because UE j may not lie in the coverage area of all the SBSs (however, we consider that the MBS is always in $\mathcal{A}_j$). Moreover, we assume that each BS can support at most N_i UEs simultaneously and that a UE is associated to exactly one BS at a time. Each BS i communicates to its served UEs using equal transmit power. In addition, we assume that the mmWave SBS allocates the entire available band to their served UEs, whereas the MBS equally shares its band across its UEs. Finally, we assume that there exists a central controller, collocated with the MBS, able to collect and forward information to the UEs in the network.

Channel Model. For simplicity of analysis, and since we consider a dense regime, we denote with R_0 the size of the coverage range of SBSs. Thus, a UE can only be associated with a SBS located at most at a distance R_0. Moreover, we consider that each communication link experiences a small scale m-Nakagami fading. We use h to denote the fading coefficient, which follows a normalized Gamma distribution $\Gamma\left(m, \frac{1}{m}\right)$. We assume Rayleigh fading for UE-MBS links, which is a special case of m-Nakagami, where $m = 1$. In addition, we adopt the commonly used Friis propagation loss model [TB15], where the received power P^{Rx} is given as a function of the distance d between the UE and its serving BS:

$$P^{Rx}(d) = hP_s^{Tx}G_s^{Tx}G_s^{Rx}C_s d^{-\eta_s}, \quad s \in \{\mathrm{SBS}, \mathrm{MBS}\}.$$

Here, C_s denotes the path-loss constant, η_s is the path-loss exponent, and P_s^{Tx} is the transmit power w.r.t. BS s. The gains of the transmitter and receiver antennas w.r.t. BS s are G_s^{Tx} and G_s^{Rx} respectively. In addition, we assume that the UEs and the BSs perform beam steering in advance such that when a communication is set up, the useful received power in absence of interference is maximized, i.e., $G_s^{Tx} = G_{max}^{Tx}$ and $G_s^{Rx} = G_{max}^{Rx}$ are the maximum antenna gain at the transmitter and the receiver, respectively.

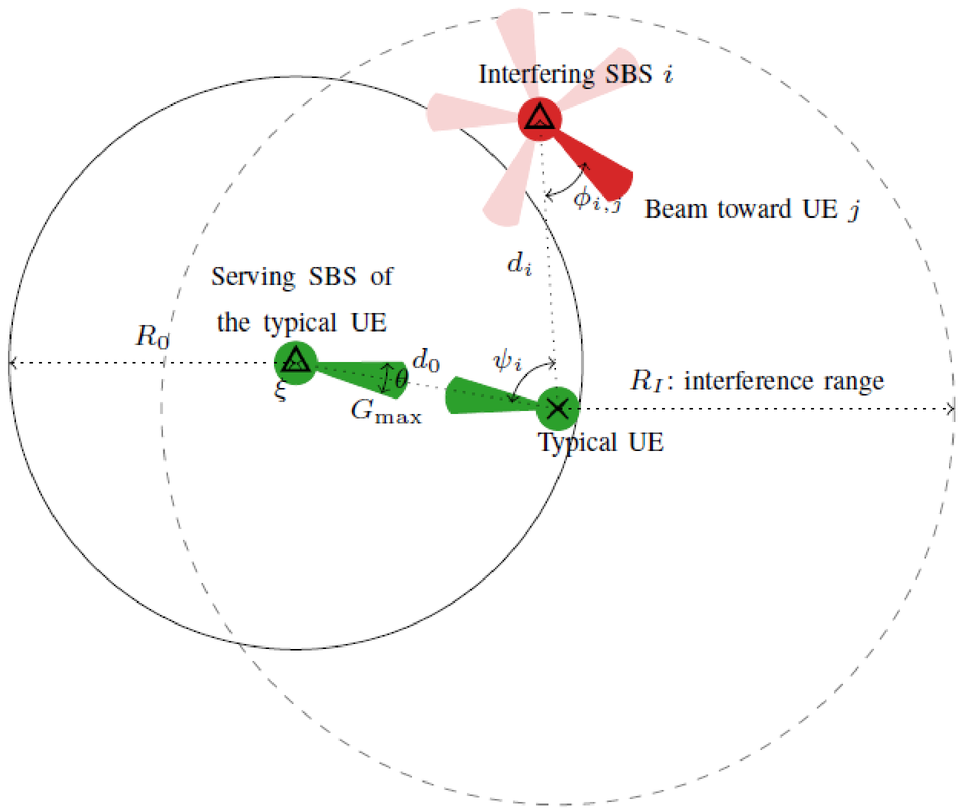


Figure 3.3-2 – Cell interference illustration.

Cell interference. Since we assume the presence of only one MBS, which orthogonalizes its served UEs by sharing its band across them, the communication links between the MBS and its served UEs experience neither intra-cell nor inter-cell interference. Therefore, interference is only due to the communications between mmWave SBSs and UEs, as a result of overlapping beams. Let us consider a typical UE (say UE j_0) placed at a distance d_0 from its serving SBS (say SBS i_0). Given an interfering SBS i that is located at a distance d_i, and relative angle ψ_i from this UE, serving $n_i \leq N_i$ other UEs in n_i random directions defined by their relative angle $\phi_{i,j}$ (see Figure 3.3-2), we use $I_{i,j}$ to denote the interference caused by its j-th beam towards the typical UE. Thus, $I_i = \sum_{j=1}^{n_i} I_{i,j}$ is the total interference incurred by this SBS on the typical UE. Moreover, for sake of simplicity, we assume that the receiver and the transmitter use the same antenna radiation pattern denoted by $G(\theta, \alpha)$, where θ is the beam width and α is the azimuthal angle to the main lobe (see Figure 3.3-2). Hence, the interference induced by the communication between the i-th SBS and its j-th UE and the total interference resulting from SBS i are:

$$I_{i,j} = P^{Tx} h_i G(\theta, \psi_i) G(\theta, \phi_{i,j}) C d_i^{-\eta},$$

$$I_i = P^{Tx} h_i C \sum_{j=1}^{n_i} G(\theta, \psi_i)\, G(\theta, \phi_{i,j}) C d_i^{-\eta}.$$

Thus, the signal-to-interference plus noise ratio SINR_{i_0,j_0} between the typical UE j_0 BS i_0, which comprises both intra-cell and inter-cell interference is defined as:

$$\mathrm{SINR}_{i_0,j_0} = \frac{P^{Tx} h_0 G_{max}^2 C d_0^{-\eta}}{\sum_{i \in \mathcal{A}} I_i + N_0 B_{i_0,j_0}},$$

where $B_{i,j}$ is the bandwidth allocated to the UE j by BS i, and N_0 is the Gaussian noise power spectrum density.

3.3.3. Problem Formulation

At time t, each UE j requests a minimum data rate $D_j(t)$ from its serving BS i to satisfy a certain quality of service (QoS), called here the UE's data demand. We say that UE j's QoS is fully satisfied at time t, if the achievable data rate $R_{i,j}(t) = B_{i,j} \log_2\left(1 + \mathrm{SINR}_{i,j}(t)\right)$, given by the Shannon capacity of the channel between UE j and BS i, is greater than $D_j(t)$, i.e., $R_{i,j}(t) \geq D_j(t)$. Based on this, we consider that at each time t, the effective communication rate between UE j and its serving BS equals $\min\left(R_{i,j}(t), D_j(t)\right)$. Hence, we can define an α-fair network utility function [RS14] as follows:

$$R(t) = \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}\, U_\alpha\left(\min\left(R_{i,j}(t), D_j(t)\right)\right)$$
$$R(t) = \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}\, U_\alpha\left(\min\left(1, \frac{R_{i,j}(t)}{D_j(t)}\right) D_j(t)\right)$$
$$R(t) = \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}\, U_\alpha\left(\kappa_{i,j}(t) D_j(t)\right),$$

where $x_{i,j} = 1$ indicates that UE j is associated with BS i; otherwise $x_{i,j} = 0$ and $\kappa_{i,j}(t) = \min\left(1, \frac{R_{i,j}(t)}{D_j(t)}\right) \in [0,1]$ indicates the QoS satisfaction of UE j w.r.t. its associated BS i, which is fully satisfied when $\kappa_{i,j} = 1$. $U_\alpha(\cdot)$ is the α-fair utility function given in [RS14] as follows:

$$U_\alpha(x) = \begin{cases} (1-\alpha)^{-1}x^{1-\alpha}, & \text{for } \alpha \geq 0 \text{ and } \alpha \neq 1 \\ \log(x), & for\ \alpha = 1 \end{cases}.$$

Then, we formulate the user association problem to maximize the network utility $R(t)$ at time t as follows:

$$\begin{aligned} \max_{\{x_{i,j}\}} R(t) = & \sum_{i\in\mathcal{A}}\sum_{j\in\mathcal{U}} x_{i,j}\, U_\alpha\left(\kappa_{i,j}(t) D_j(t)\right) \\ \text{subject to:}\quad & x_{i,j} \in \{0,1\}, \qquad \forall i,j \\ & \sum_{j\in\mathcal{U}_i} x_{i,j} \leq N_i, \qquad \forall i \in \mathcal{A}\backslash\{0\} \\ & \sum_{i\in\mathcal{A}_j} x_{i,j} = 1, \qquad \forall j \in \mathcal{U}. \end{aligned}$$

The first constraint indicates that the $x_{i,j}$ are binary variables. The number of resources available at each SBS is limited; this is highlighted in the next constraint, by constraining the number of UEs simultaneously associated with a given SBS i, to be lower than N_i. Finally, the third constraint ensures that, in our setting, each UE is associated with exactly one BS.

Depending on the value of α, this optimization problem guarantees different fairness criteria in the user association.

Lemma 1. *When $\alpha = 0$, $U_\alpha(x) = 1$ and the optimization problem is equivalent to the network sum-rate maximization problem. When $\alpha \to 1$, $U_\alpha(x)$ converges to the proportional fair utility function as $\lim_{\alpha\to 1} U_\alpha(x) = log(x)$ and the optimization problem falls into network sum-log-rate maximization, also known as proportional fairness user association. As $\alpha \to \infty$, $U_\alpha(x)$ approaches max-min fairness utility function and the optimization problem is equivalent to max-min fairness user association.*

Proof. Proof can be found in [RS14], Section 2.2.1. ■

In the following, we will focus on $\alpha \in \{0,1\}$, widely used in the literature [DL16], [AA19a-b].

3.3.4. Proposed Transferable User Association Policy

Taking into account the targeted optimization objective, we derive in this section an adaptive association policy capable of solving the user association problem regardless of the location and the number of UEs in the network. The desired policy must be able to adapt to the departure or arrival of UEs from and in the network, as both events have an impact on the optimal user association. To do so, we propose to construct a transferable policy neural network architecture, invariable with the number of UEs, which can be efficiently trained and then transferred to any UE that arrives in the cell. This policy leverages UEs' local information and possibly global information, e.g., from a central controller, which without loss of generality we can assume to be collocated with the MBS, to optimize the association decisions using a MARL framework.

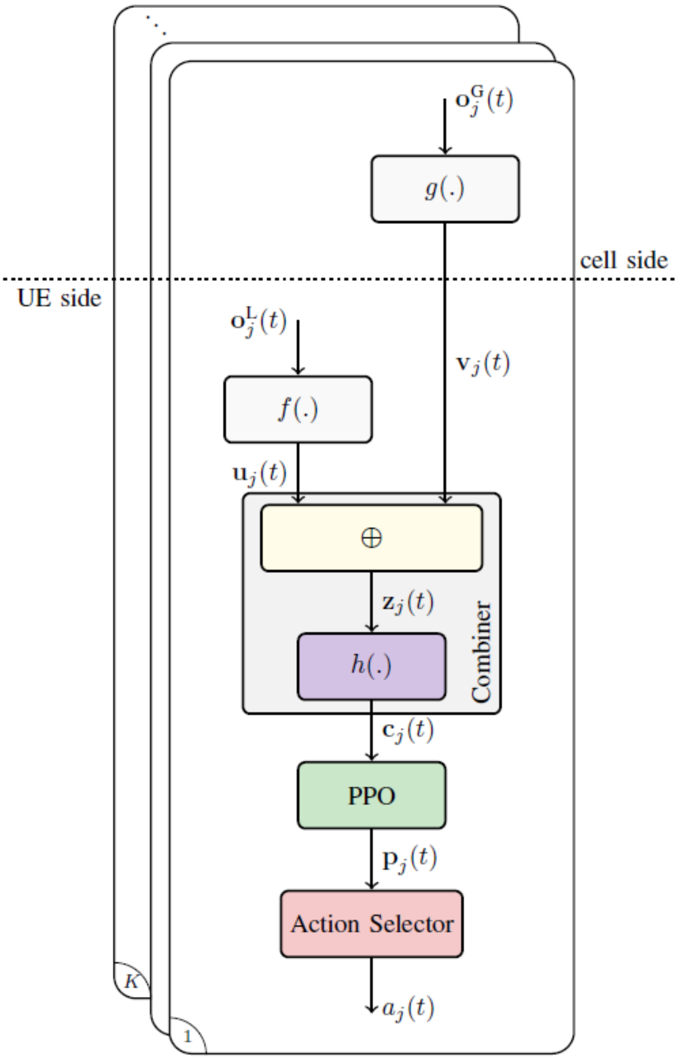


Figure 3.3-3 – UE association policy network architecture (PNA). The model is the same for UEs from 1 *to K.*

General framework

In this section, we provide a general description of the PNA, whose component design details will be specified in the following Section. For now, let us denote by $\mathbf{o}_j^{\mathrm{L}}(t)$ and $\mathbf{o}_j^{\mathrm{G}}(t)$ the *local* and *global* observation of UE j respectively. $\mathbf{o}_j^{\mathrm{L}}(t)$ comprises the set of measurement signals directly accessible to (or measurable by) the user's device. Instead, depending on the optimization objective and constraints, $\mathbf{o}_j^{\mathrm{G}}(t)$ embeds higher-level information (macro observations), which can be collected and forwarded to UEs by the central controller. Then, in our proposed framework, each UE starts by building its local state encoding $\mathbf{u}_j(t) = f\left(\mathbf{o}_j^{\mathrm{L}}(t)\right)$ and global state encoding $\mathbf{v}_j(t) = f\left(\mathbf{o}_j^{\mathrm{G}}(t)\right)$ using differentiable and learnable functions $f(\cdot)$ and $g(\cdot)$[1] (i.e., functions with learnable parameters like NNs). Next, the local and global state encoding are combined together to form the agent context encoding $\mathbf{c}_j(t)$ using a combiner function $h(\cdot)$. The role of this combiner function, which is also chosen to be differentiable and learnable, is to build UE context understanding vector, as a representation of its local and global observations. Now, given the context vector $\mathbf{c}_j(t)$, the goal of the learning agent j at each time

[1] One can view this process as a filtering stage, which consists in building a state representation of the input observations.

instant t, is to define an *association probability vector* $\mathbf{p}_j(t) = [p_{1,j}, \dots, p_{N_s+1}] \in [0,1]^{N_s+1}$with $\sum_{i\in\mathcal{A}} p_{i,j} = 1$ and $p_{i,j} = 0$ for all $i \notin \mathcal{A}_j$. Then, UE's action $a_j(t)$, which corresponds to a connection request towards the BS indexed by $a_j(t)$ in $\mathcal{A}_j$, is sampled from the distribution characterized by the $p_{i,j}$. Thus, the learning problem here consists in deriving an association policy that optimizes the corresponding association probability vector $\mathbf{p}_j(t)$, so that sampling from it maximizes the network utility function $R(t)$. Figure 3.3-3 describes the proposed PNA. Note that in this architecture, UEs' agents share the same model, i.e., $f(\cdot)$, $g(\cdot)$, and $h(\cdot)$ are common to all UEs. This setting does not preclude UEs from taking different actions as they do not observe the same inputs. In contrast, sharing the parameters among UEs enables a better skill transfer since there is only a unique policy (in contrast to having one policy per UE as in [NZ18]), which can be efficiently and simultaneously trained with the experiences of all UEs in a MARL framework using PPO [JS17].

Proximal Policy Optimization (PPO). In a MARL system, agents learn by interacting with a shared environment by making decisions following a Markov Decision Process (MDP). In MDP, the action $a_j(t)$ of an agent j in a given state $\mathbf{s}_j(t)$ leads it to the next state $\mathbf{s}_j(t+1)$ and results in a reward $r_j(t)$. From the underlying *experience* $e_j(t) = \{\mathbf{s}_j(t), a_j(t), r_j(t), \mathbf{s}_j(t+1)\}$, the agent learns its policy $\pi_{j,\mathbf{w}}(\cdot \mid \cdot)$, parameterized by **w**, the set of PNA parameters, where $\pi_{j,\mathbf{w}}(a_j|\mathbf{s}_j)$ is the probability that agent j takes action a_j in state $\mathbf{s}_j$[2], to maximize an accumulated long-term γ-discounted reward $G_j(t) = \sum_{\tau=t+1}^{T_e} \gamma^{\tau-t-1} r_j(t)$ over an *episode* – a new network deployment – of duration T_e:

$$\pi_{j,\mathbf{w}}^* = \arg\max_{\pi_j} \mathbb{E}_t\,[G_j(t)].$$

In our study, we consider the particular case of cooperative MARL [LB08], i.e., UEs share the same reward, hence, they are assigned to the same objective of maximizing the network utility function: $r_j(t) = R(t)$. Moreover, UEs also share the same policy, i.e., $\pi_{j,\mathbf{w}} = \pi_{\mathbf{w}}$ for all j.

In general MARL, an agent has only access to a partial observation $\mathbf{o}_j(t) = \{\mathbf{o}_j^{\mathrm{L}}(t), \mathbf{o}_j^{\mathrm{G}}(t)\}$ of the actual state $\mathbf{s}_j(t)$, which is unknown, resulting in partially observable MDP (POMDP) [SO17]. Moreover, MARL is subject to non-stationarities due to simultaneous interactions of agents with the environment, which make the learning process more complex. In the literature, policy gradient algorithms are used to solve this problem [RSS18].

We use an actor-critic mechanism to iteratively update the policy parameters **w** to minimize the ϵ-clipped proximal loss function:

$$\mathcal{L}(\mathbf{w}) = \mathbb{E}_\pi\big[\min\big(\zeta(\mathbf{w})\hat{A}, \mathrm{clip}(\zeta(\mathbf{w}), 1-\epsilon_1, 1+\epsilon_2)\hat{A}\big)\big],$$

where $\mathrm{clip}(x,a,b) = \min(\max(x,a),b)$, $\hat{A}(a_j, \mathbf{o}_j)$ denotes the advantage estimator, which measures the advantage of selecting a given action in a given state, that we estimate using one step Temporal Difference error [RSS18]. $\zeta(\mathbf{w}) = \frac{\pi_{\mathbf{w}}(a_j|\mathbf{o}_j)}{\pi_{\mathbf{w}_{\mathrm{old}}}(a_j|\mathbf{o}_j)}$ is the probability ratio between the current and previous update.

By introducing the clipping effect, PPO pessimistically ignores updates (possibly destructive) that will lead to high changes between policy updates.

Hysteretic PPO. Note that in vanilla PPO, $\epsilon_1 = \epsilon_2$. However, in multi-agent environments, an agent should not be pessimistic in the same way for both *positive* ($\zeta(\mathbf{w}) > 1$) and *negative* ($\zeta(\mathbf{w}) < 1$). In fact, due to the interaction of multiple agents with the environment and the common reward of the cooperative framework, an agent may receive a lower reward because of the bad behaviour of its teammates. This may cause the user to change its policy at the risk to misleading it. To overcome this issue, following the concept of hysteretic Q-learning in [LM07], we introduce *hysteretic proximal policy optimization*, where we use ϵ_1 for negative updates and ϵ_2 for positive updates, such that $\epsilon_1 < \epsilon_2$. In this way, an agent gives more importance to updates that improve its policy rather than to ones that worsen it. This setting is particularly important when agents do not have equal contribution to the team's reward and for decentralized learning.

Notice that the association policy can be efficiently trained in a centralized way with the experience of all agents. In case a decentralized approached is envisaged, this can be obtained by leveraging the decentralized and distributed PPO approaches presented in [EW20].

[2] Note that $\pi_{j,\mathbf{w}}(a_j|\mathbf{s}_j) = p_{a_j(t),j}$, where $p_{k,l}$ is the association probability.

On Transferable Policy Architecture: PNA Components Design

In order for the policy architecture to be transferable, an adequate design of the PNA components is required. Our objective, in fact, is to construct a policy architecture whose size does not vary with the number of UEs in the network, which is bound to change over time. In the following, we will describe the main components of PNA, including the contents of local and global observations, as well as the characteristics of encoding functions $f(\cdot)$, $g(\cdot)$, and $h(\cdot)$, which allow the transferability of the policy architecture.

UE Local Observation Encoding. In this study, we assume that at each time step, each UE j can estimate the received signal strength (RSS) and the corresponding angle of arrival (AoA) w.r.t. its surrounding BSs. We denote with $\mathrm{RSS}_{i,j}$ and $\vartheta_{i,j}$ the estimated RSS and AoA of UE j w.r.t. BS i, respectively. Moreover, as in [MS20], a UE receives an acknowledgment (ACK/NACK) signal whenever its connection request succeeds ($\mathrm{ACK}_j = 1$) or is denied ($\mathrm{ACK}_j = 0$), which may happen due to the limited resources available at each BS (the fact that N_i in the problem formulation is given and finite); we call this event a collision. Hence, we define the local state of a UE, $\mathbf{o}_j^{\mathrm{L}}(t)$, as follows:

$$\mathbf{o}_j^{\mathrm{L}}(t) = \left\{a_j(t-1), R_{a_j(t-1),j}, R(t-1), \mathrm{ACK}_j, \left\{\mathrm{RSS}_{i,j}\right\}_{i\in\mathcal{A}_j}, \left\{\vartheta_{i,j}\right\}_{i\in\mathcal{A}_j}\right\}.$$

Here, $R_{a_j(t-1),j}$ represents the achievable communication rate when UE j is associated with the BS indexed by $a_j(t-1)$.

Note that the size of $\mathbf{o}_j^{\mathrm{L}}(t)$ does not depend on the number of UEs, in sharp contrast with [NZ18]. Then, we obtain the n-dimensional local encoding vector $\mathbf{u}_j(t) = f\left(\mathbf{o}_j^{\mathrm{L}}(t)\right)$, where $f : \mathbb{R}^l \to \mathbb{R}^n$ is a neural network, and l is the size of the vector obtained after the concatenation of the elements in $\mathbf{o}_j^{\mathrm{L}}(t)$.

Remark 1 (Collision events handling)**.** *Note that collisions may occur when a BS receives more connection requests than it can support. In previous work on static user association* [MS20]*, we severely discouraged collisions by zero-rewarding UEs when collision events occurred; however, here, as the positions of UEs change from one episode to another, the collision management is considerably more complex. Agents have to learn that collision events do not only depend on their actions but also on their relative positions. To handle such a complexity, we consider a softer solution: when a collision occurs, the BSs send a NACK signal to notify UEs of collision, then each BS selects among the colliding UEs the best ones to associate with, according to their association probability. In this way, the reward of UEs is not severely zeroed and we directly relate the collision events to the training performance.*

UE Global Observation Encoding. After taking an action $a_j(t)$, the controller can encode for UE j some meaningful global state (i.e. macro) information $\mathbf{o}_j^{\mathrm{G}}(t)$ such as the estimated position of UEs of interfering links, i.e., of active mmWave links, the load of each BS, etc. However, one has to notice that embedding more information does not necessarily imply performance improvement as it also increases the agent's state space, thus requiring more exploration to discover the intrinsic state/action relation at the risk of misleading the agent. In our scenario, we consider that the information about the actual rate perceived by each UE j and the position of the potential interferers of UE j, i.e., the set of UEs $\mathcal{N}_j(t)$, susceptible to impact the association decision of UE j through the interference resulting from their communications[3]. Hence, we define $\mathbf{o}_j^{\mathrm{G}}(t)$:

$$\mathbf{o}_j^{\mathrm{G}}(t) = \left\{\varsigma_l = \left[x_l, y_l, R_{a_l(t-1),l}\right], l \in \mathcal{N}_j\right\}.$$

As $\mathcal{N}_j(t)$ varies with time, constructing UE j's global state encoding vector $\mathbf{v}_j(t) = g\left(\mathbf{o}_j^{\mathrm{G}}(t)\right)$ is very challenging.

A naive solution to construct $\mathbf{v}_j(t)$ is to first concatenate all elements in $\mathbf{o}_j^{\mathrm{G}}(t)$, resulting in a vector of size $m = 3 \times \mathrm{card}\left(\mathcal{N}_j\right)$. Then, we obtain the local encoding vector $\mathbf{v}_j(t) = g\left(\mathbf{o}_j^{\mathrm{G}}(t)\right)$, where $g: \mathbb{R}^m \to \mathbb{R}^n$ is also a NN. However, such an approach

- has limited scalability, as the size of $\mathbf{o}_j^{\mathrm{G}}(t)$ varies with the number of UEs, especially in the neighborhood, and
- requires ordering elements prior to concatenation, preventing from transferability.

[3] Note that in this study, and for the sake of simplicity, we consider $\mathcal{N}_j$ as the k-nearest neighbors of UE j however, solutions based on local interaction graphs can be considered, where potential interferers can be identified based on an interference threshold following approaches in **Erreur ! Source du renvoi introuvable.**.

Therefore, an efficient solution to the problem should be agnostic of the ordering in $\mathbf{o}_j^{\mathrm{G}}(t)$. Moreover, in order to build a scalable and transferable architecture, the size of $\mathbf{v}_j$ should be also independent of the length of $\mathbf{o}_j^{\mathrm{G}}(t)$, i.e., the number of UEs in UE j's neighborhood. To satisfy these properties, we adopt ideas from the *dot-product attention* mechanisms developed in [AV17]. Considering this approach, let $\mathbf{k}_j = g_k(\varsigma_j)$, $\mathbf{q}_j = g_q(\varsigma_j)$, and $\mathbf{v}_j = g_v(\varsigma_j)$, where $g_k, g_q, g_v : \mathbb{R}^3 \to \mathbb{R}^n$ are also encoding functions (e.g., neural networks), and $\mathbf{k}_j$, $\mathbf{q}_j$, and $\mathbf{v}_j$ denote the *key*, the *query* and the *value* associated with UE j, respectively. For a given UE j, we compute for each UE in its neighborhood $\mathcal{N}_j$ a weight (or score) $\alpha_{k,j}$:

$$\boldsymbol{\alpha}_j = [\alpha_{k,j}] = \mathrm{softmax}\left(\left[\frac{\mathbf{q}_k \mathbf{k}_j^T}{\sqrt{n}}\right]_{k\in\mathcal{N}_j}\right).$$

Here, $\mathrm{softmax}(\cdot)$ is the softmax function also known as the normalized exponential function. Let $\boldsymbol{\alpha}_j = [\alpha_{k,j}, k \in \mathcal{N}_j]$. The vector $\boldsymbol{\alpha}_j$ represents the interaction of UE j with its neighbors. Then, we compute the encoding $\mathbf{v}_j$ by aggregating all values' information from the neighborhood as follows:

$$\mathbf{v}_j = \sum_{k\in\mathcal{N}_j} \alpha_{k,j}\, \mathbf{v}_k.$$

Remark 2. *This process can also be viewed as a message passing between UEs. In this case, UEs only need to exchange their queries and values with each other in the neighborhood.*

Remark 3. *By construction, the size of $\mathbf{v}_j$ built as a linear combination of the $\mathbf{v}_k$ is invariable with the size of $\mathcal{N}_j$. That is to say, whenever the number of UEs varies, there is no need to change the PNA.*

Now, once we obtain the UE local and global encoding vector, they are merged together to build its context understanding vector, i.e., its perception of the radio environment $\mathbf{c}_j(t) = h\left(\mathbf{z}_j(t)\right)$, where $\mathbf{z}_j(t) = \mathbf{u}_j(t) \oplus \mathbf{v}_j(t)$. . This is done thanks to the combiner function $h : \mathbb{R}^{2n} \to \mathbb{R}^n$, which we consider here to be NN.

Finally, note that the association policy can be efficiently trained in a centralized way with the experience of all agents or in a decentralized way, by leveraging approaches presented in [EW20]. To further make learning robust against the variability of the number of UEs over time, we introduce a UE dropout mechanism with rate p_0, corresponding to the Bernoulli probability of a UE to be masked out in a given training episode, thus, appearing as non-existent in the cell for the others UEs.

3.3.5. Numerical results

In this section, we evaluate the effectiveness of our approach in different simulation setups. We assess both the impact of the learning parameters and radio environment dynamicity on the system performance. We also evaluate the zero-shot generalization capacity of the proposed framework and hence, its transferability.

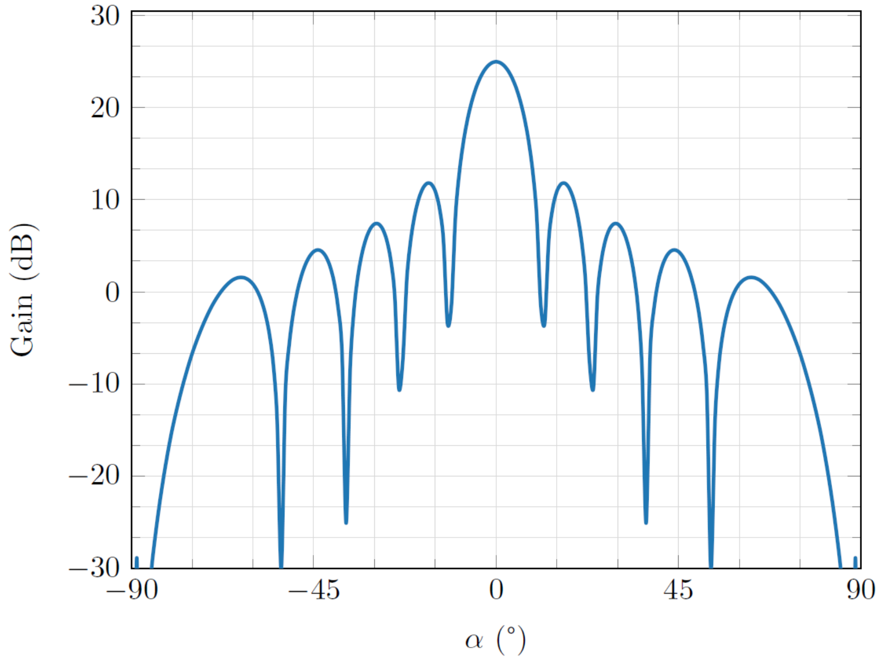


Figure 3.3-4 – Simulated TX/RX antenna gain radiation pattern for an array of 10 by 10 elements operating at 28 GHz [RJM17].

Radio environment. In our simulations, we consider $K_0 = 15$ UEs randomly located in a bi-dimensional region, under the coverage of $N_s = 3$ SBSs working at mmWave frequencies with a carrier frequency of 28 GHz, and one MBS communicating at 2 GHz. We assume that when UEs and SBSs communicate together, they use the same antenna radiation pattern obtained through the analog beamforming shown in Figure 3.3-4. In contrast, the MBS transmits via a 17-dBi omnidirectional antenna. In addition, we assume that the error in the estimation of the AoA follows a normal distribution with mean equal to 2°. In addition, in our simulations, we consider three types of service corresponding to an average data rate demand $\overline{D_j} \in \{5{,}200{,}1500\}$ Mbps. We assume that the traffic request of a UE j is a random variable, which follows a Poisson distribution with intensity $\overline{D_j} = \mathbb{E}\big[D_j(t)\big]$. Additional simulation parameters can be founded in Table 3.3-1. Note that we compute the path-loss constant as $C_s = (c/4\pi f_s)^2$, where f_s is the carrier frequency and $c = 3 \times 10^8$ ms^{-1} is the speed of light.

UE action space. Since all UEs share the same policy network, $\mathcal{A}$ coincides with the action space. In this way, we guarantee a fixed action space for all UEs irrespective of their positions. However, a UE j can only be associated with BSs in $\mathcal{A}_j \subseteq \mathcal{A}$. Accordingly, unauthorized actions or connection requests $a_j(t) \notin \mathcal{A}_j$ are redirected towards the MBS, i.e., they appear as connection requests to the MBS.

Learning parameters specification. We fixed the size of the encoding functions $n = 128$. All encoding functions are composed of only one hidden multi-layer perceptron (MLP) of n neurons. The network parameters are optimized using actor-critic PPO [JS17], where both actor and critic comprise also one hidden layer with $2n$ neurons. All layers use a rectifier linear unit (ReLU) activation. We set the learning rate μ to 10^{-4} and the discounting factor $\gamma = 0.6$. Unless specified, we empirically fix the clipping factors to $\epsilon_1 = 0.01$, $\epsilon_2 = 0.5$, the time horizon to $T_e = 250$ and the dropout probability to $p_0 = 0.95$. Also, we limit the neighborhood of a UE to its k-nearest neighbors, where $k \leq 15$.

Benchmarks. As a comparison, we consider the same benchmarks as in [MS20], i.e., the Max-SNR algorithm, which associates UEs on the basis of links with the maximum SNR, and the centralized heuristic algorithm, which consists in associating UEs, starting from the links with the maximum SNR, and in an iterative way as long as it increases the network utility. Originally proposed in [PZ19], the centralized heuristic algorithm is shown to exhibit good performance, specifically in interference-limited network. Therefore, we use it as a baseline solution in lieu of the optimal solution, infeasible here, due to the network size.

To assess the convergence performance of the proposed algorithm, we define $r_d(t) = \bar{R}^{Trans.\ RL}(t) - \bar{R}^{Heur.}(t)$, which corresponds to the difference of the average reward over an episode reached by the proposed algorithm compared to the centralized heuristic approach. For sake of clarity, we plot the associated rolling average and standard deviation on a 100-sized window, with a logarithmic scale on the x-axis. Also, unless otherwise specified, we represent on the histograms, the average performance over 500 random deployments of UEs.

Table 3.3-1. Simulation parameters [MS20].

PARAMETERS	VALUES (MACRO CELL)	VALUES (SMALL CELL)
CARRIER FREQUENCY, f_s	2 GHz	28 GHz
BANDWIDTH	10 MHz	200 MHz
THERMAL NOISE, N_0	-174 dBm/Hz	-174 dBm/Hz
NOISE FIGURE	5 dB	0 dB
SHADOWING VARIANCE	9 dB	12 dB
TX POWER, P^{Tx}	46 dBm	20 dBm
ANTENNA GAIN, G^{Tx}_{max} / G^{Rx}_{max}	17 dBi / 0 dBi	(see radiation pattern)
RADIUS, R_0		50 m
BACK-LOBE GAIN		-20 dB
PATH-LOSS EXPONENT, η_s	3.76	2.5
INTER-CELL DISTANCE		$1.2 \times R_0$

Convergence Properties.

In this section, we aim to evaluate the algorithm's convergence w.r.t. the aforementioned learning parameters.

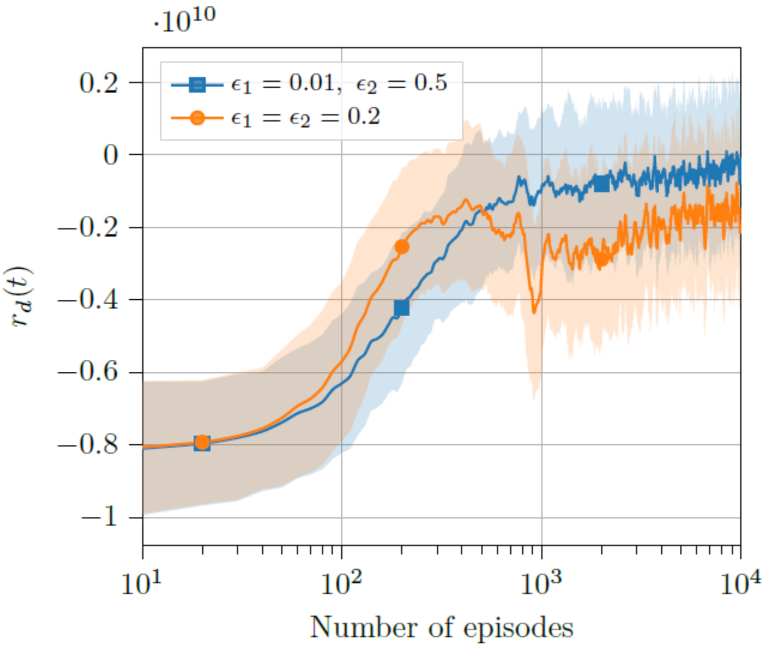


Figure 3.3-5 – Effect of the hysteretic clipping factors on the convergence for network-sum rate maximization.

Effect of Hysteretic Clipping Factors on Convergence. Let us start by evaluating the impact of clipping factors ϵ_1 and ϵ_2 on the convergence. Figure 3.3-5 is obtained fixing $\alpha = 0$ in the optimization function and $D_j(t) = \infty$ for all j. It shows the evolution of $r_d(t)$ in two settings: $\epsilon_1 = \epsilon_2 = 0.2$, corresponding to the setting of the vanilla PPO proposed in [JS17], and our empirically optimized hysteretic setting $\epsilon_1 = 0.01,\ \epsilon_2 = 0.5$. We show that by simply introducing a hysteretic effect in the clipping factors, we notably improve the stability and the learning performance, reaching the same performance as the heuristic algorithm (as $r_d(t)$ converges on average to zero).

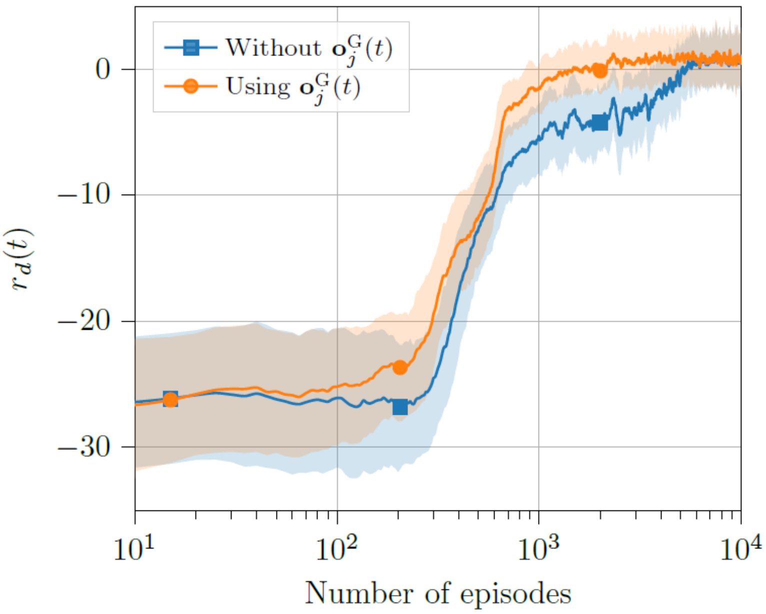


Figure 3.3-6 – Global information impact on network convergence for network-sum log-rate maximization.

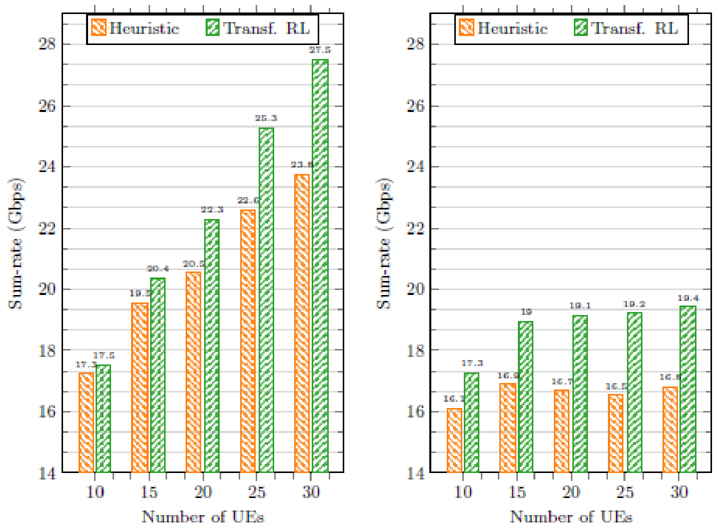
Impact of Global Information on Convergence. Here, we assess the add-on impact of the global information $\mathbf{o}_j^{\mathrm{G}}(t)$ for the learning convergence. Figure 3.3-6 is obtained by fixing $\alpha = 1$ and $D_j(t) = \infty$ for all j and shows the evolution of $r_d(t)$ when UEs have or do not have access to global information[4]. We can remark that $\mathbf{o}_j^{\mathrm{G}}(t)$ can effectively help accelerate the convergence of the algorithm. However, after 5×10^3 episodes, the two curves eventually end up with the same performance. This last result comes from the fact that the information (i.e., $\varsigma_k, k \in \mathcal{N}_j$) carried on $\mathbf{o}_j^{\mathrm{G}}(t)$ is also embedded in $\mathbf{o}_j^{\mathrm{L}}(t)$ through the RSSI and $R(t)$, although this information is "drowned". By separating each piece of information in $\mathbf{o}_j^{\mathrm{G}}(t)$, we further improve UEs' context understanding, thus the learning speed.

Policy Transferability Property

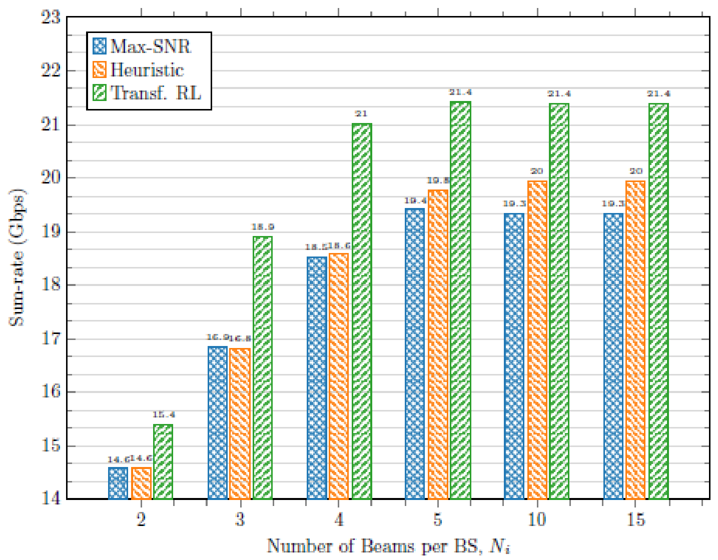
To assess how transferable the proposed algorithm is, we consider training the PNA for a reference number of users, $K_0 = 15$ and for fixed number of beams per BS, $N_i = 3$ for all i. Then, we evaluate the performance of the algorithm for different network deployments with a variable number of UEs $K \in \{10, 15, 20, 25, 30\}$, including changes in the UEs' position.

Zero-shot Generalization Capacity. To evaluate the generalization capability of the proposed algorithm, we train the PNA to optimize the network sum-log-rate, i.e., $\alpha = 1$.

[4]Note that here, the optimization is done for $\alpha = 1$; however, the same results applied to $\alpha = 0$.



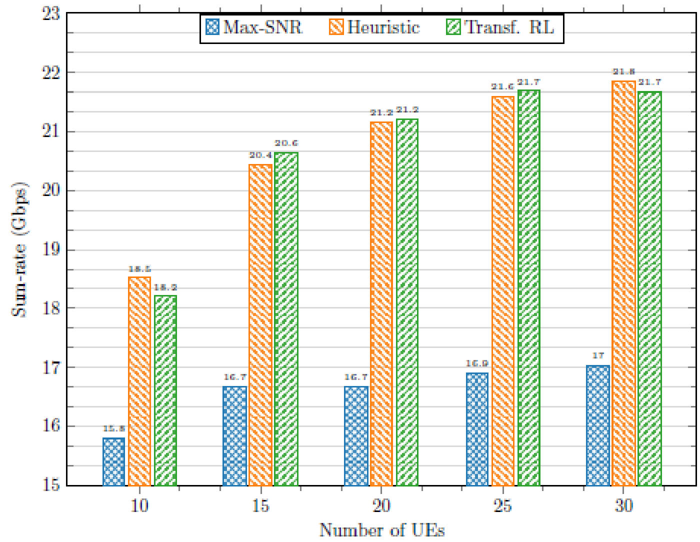
(a) $N_i = 15$ in left figure (i.e., less collision events), $N_i = 3$ in right figure. K varies.



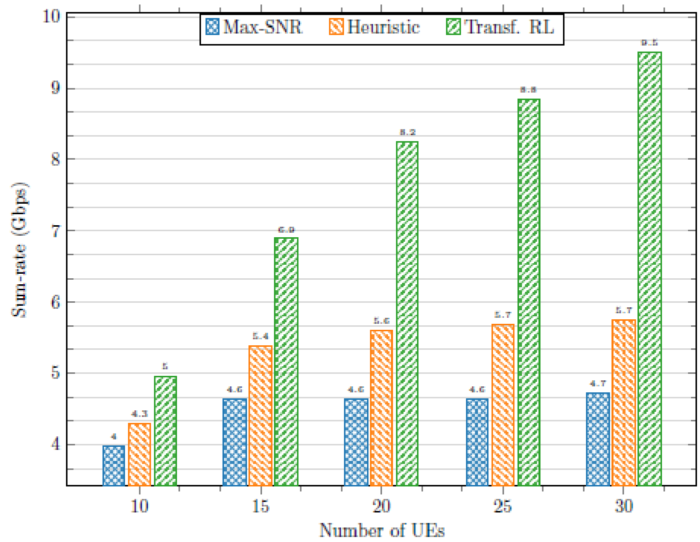
(b) Training configuration: ($K_0 = 15$ UEs, $N_i = 3$, $\forall i$). Testing configuration: K is kept fixed and equal to K_0, N_i varies.

Figure 3.3-7 – Generalizability of the proposed PNA to different numbers of UEs.

We remark that the proposed architecture can effectively and efficiently adapt to change in the number of UEs and the number of beams available per BS, without requiring additional training steps. Figure 3.3-7 is obtained with $K_0 = 15$ and $N_i = 3$ for all i. In Figure 3.3-7 (a), we can see that when the number of UEs doubles w.r.t. the reference training point from 15 to 30, the proposed transferable a PNA exhibits a 15.5% increase in network sum-rate compared to the heuristic approach. Moreover, an additional feature of the proposed architecture is that even when the number of beams available per BS later changes, which impacts the collision events, the algorithm still adapts to maintain the system's performance. Indeed, in Figure 3.3-7 (b), where we evaluate the performance of the algorithms for different $N_i \in \{2, 3, 4, 5, 10, 15\}$, we can observe that as N_i increases, implying less and less collisions since K is fixed, the algorithm keeps outperforming the two benchmarks. When N_i becomes greater than 5, i.e., $\sum_{i=1}^{3} N_i > K = 15$, there is no improvement in the sum-rate as there are enough beams to serve all UEs.



(a) Full-buffer traffic.



(b) With dynamic traffic.

Figure 3.3-8 – Performance w.r.t. network traffic.

Performance w.r.t. Network Traffic. Now we evaluate the system performance w.r.t. network traffic. Here again, the PNA is trained for $K_0 = 15$ to optimize the network sum-rate ($\alpha = 0$). Figure 3.3-8 (a) shows the case of full-buffer traffic (i.e., $D_j(t) = 1$) and Figure 3.3-8 (b) the case of dynamic traffic. We remark that in case of full-buffer traffic, the proposed method performs better than the two benchmarks but performs slightly worse than the heuristic algorithm when generalized to $K = 10$ and $K = 30$. However, when we consider the network traffic, the proposed transferable solution clearly outperforms the two benchmarks, yielding a 102.1% and a 66.66% network sum-rate increase for $K = 30$, w.r.t. the max-SNR and the heuristic algorithm, respectively.

3.3.6. Conclusion

In this study, we investigated the problem of transferability of user association policies in dense mmWave networks. To this end, we proposed a policy network architecture and a learning mechanism that enable users to learn a robust and transferable user association policy. The latter is adapted to withstand the environment dynamics, including fast fading, evolving traffic requirements, and time varying number and position of UEs. Our proposed solution is based on deep multi-agent reinforcement learning, where agents trained by a centralized controller leverage UEs' local and possibly global observations to optimize a network utility function. With the proposed novel architecture, the learned policy has zero-shot generalization capabilities and can directly be applied to new incoming UEs, whose association decisions can be taken by the central controller without requiring additional training steps.

Numerical results show that the proposed solution can achieve large network sum-rate gains, especially when we consider network traffic and mobility, indeed, doubling the network sum-rate compared to baseline approaches available in the literature. The results of this study open new perspectives to exploit the proposed solution for user association in other scenarios, e.g., in the context of mobile base stations (such as unmanned aerial vehicles) or for satellite communications to optimize network throughput or to improve network coverage. Another field of application of the proposed ideas is user association for mobile edge computing, which involves both radio and computing resources optimization.

3.4. Evaluating Handover Performance for End-to-End LTE Networks with OpenAirInterface

3.4.1. Motivation and related work

Handover (HO) in cellular network is a mechanism that provides continuity of service to a terminal with varying radio or traffic quality by switching its connection from the serving cell to a selected neighboring cell. In regular 4G Long-Term Evolution (LTE) / 5G New Radio (NR) networks, the HO decision is made by the serving eNB, with knowledge of the different signal strengths that the connected terminal detects. The HO mechanism is inherent in contexts such as a vehicle-to-everything environment due to high the mobility of vehicles or in the factory of the future due to the blockage of moving objects, the mobility of a terminal and the prioritization of one link over another (e.g., if a specific terminal has a specific quality of service to achieve). In [TGJ19], M. Tayyab et al. did a survey on innovative LTE and 5G HO techniques but this comprehensive knowledge base does not provide any experimentation reference. In general, HO procedures clearly lack the means to experiment with them. In [HSC+15], Han et al. evaluate the performance of X2 HO (as explain in section II.A) in real world environment. They use a Commercial Off The Shelf (COTS) User Equipment (UE) such as smartphone, connected to wireshark to dissect the control messages and can track the message flow. Evaluating performance in a real world environment provides insight into where the latency bottleneck is. However, since they do not have access to the core network, most of the data they bring in on delays in the core network is inferred from procedures that are not related to the HO.

Software Defined Radio (SDR) is a paradigm that enables flexible radio systems. Indeed, in SDR systems, most of the processing is delegated to software computation (e.g., turbo code encoder/decoder [KVIT04]). In the hardware remain the procedures related to the radio (e.g., transmission, reception and ADC/DAC). OpenAirInterface [NMM+14] (OAI) is an open-source framework that aims to provide a pluggable cellular network solution using SDR boards. It implements Radio Area Network (RAN) elements, consisting of the UE and the eNodeB (eNB). On the Core Network (CN) side, OAI implements the Mobility Management Entity (MME), the Home Subscriber Server (HSS), and the Serving Gateway and Packet Data Network Gateway that are combined into a single entity (SPGW). However the Policy and Charging Rules Function (PCRF) is not implemented. OAI currently supports 4G LTE, and non standalone 5G NR limited to the use of COTS UE. In [ANKB16], Alexandris et al. present the HO in OAI, but they do not specify which layers are enabled or not. Similarly, they do not mention the presence of a CN in their experimental setup. In [MBHS20], Manco et al. are experimenting LTE V2X with OAI, using sidelink capabilities. However, there is no mention of mobility which is a major point in V2X as vehicles are more likely to travel distances not covered by a single cell.

To our knowledge, there is no experiment with the HO procedure using an accessible and reconfigurable environment with an end-to-end architecture (i.e. including both RAN and CN). An implementation of HO in OAI exists but only with COTS UE. The use of a COTS UE reduces the degree of freedom of research experimentation. Indeed, we may be limited by the implementation of standards and constructors. Having a full SDR experimentation setup allows us to bypass these limitations. For instance, if we want to evaluate a scenario in which the eNB or another entity has full control over the HO trigger and decision, we can disable the UE measurement reports to avoid unnecessary control traffic, which is not possible with a COTS UE. In this study, we address an end-to-end, configurable experimental testbed for HO procedure. The testbed is end-to-end as it involves components from a UE to the SPGW CN and is configurable as we have access to the code of any RAN and CN component and can customize it. Our contribution is threefold: first, we implemented LTE X2 HO in the OAI RAN. This includes (but is not limited to) the implementation of multi-eNB management, such as eNB synchronization eNB scanning and selection by UE RAN layers, Reference Signal Received Power (RSRP) measurement of neighboring cells, Contention-Free Random Access (CFRA) versus Contention-Based Random Access (CBRA) procedure. Next, we experimented with X2 HO for realistic mobility scenarios in an end-to-end, full-SDR environment with an accurate channel emulator. Finally, we analysed the performance in terms of end-to-end throughput and latency for each step of the procedure.

3.4.2. Architecture and procedures

In this section, we recall the LTE architecture and the Random Access (RA) procedure. We also provide an overview of the HO procedure we implement, including the HO message flow, HO conditions and HO latency decomposition.

3.4.2.1. LTE Architecture

LTE is split into two entities as illustrated in Figure 3.4-1, the evolved universal terrestrial radio access network which we refer to as RAN in this study, and the Evolved Packet Core (EPC), also referred to as CN. RAN is composed of UEs and eNBs. The link between the eNBs is called X2.

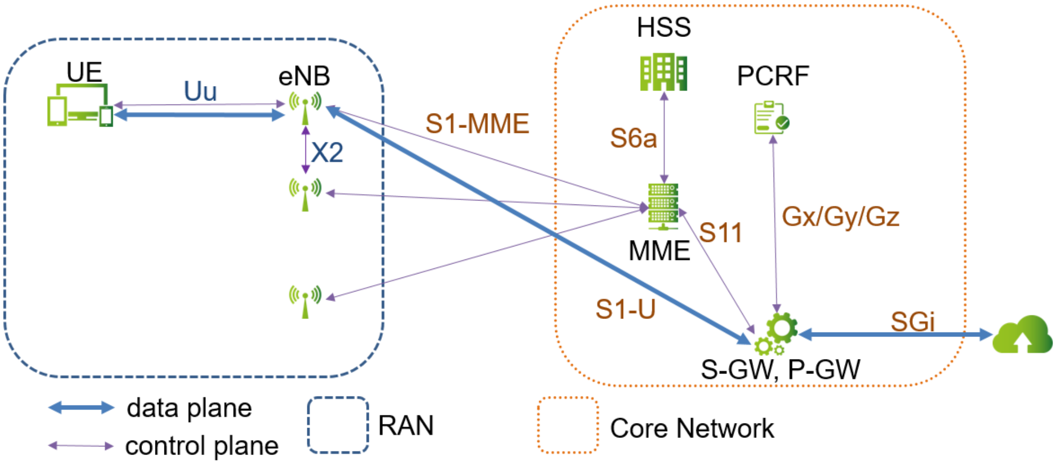


Figure 3.4-1 – LTE end to end architecture

When the UE is switched on, it searches for surrounding eNBs and selects one. It establishes the connection with the eNB by performing a RA procedure via the Physical Random Access Channel (PRACH). There are two RA procedures: Figure 3.4-2 and Figure 3.4-3 describe the CBRA and the CFRA procedures, respectively. In CBRA, the UE chooses a random number that is used by the eNB to identify the new UE. It involves additional control, as several UEs may choose the same RA preamble. In CFRA, the preamble is given by the eNB through the previous signaling. The RA in the HO can be either one or the other. In our testbed, we choose the CBRA procedure for the initial connection and the CFRA procedure for the HO because it involves less control traffic, thus reducing the HO time. The connection procedure ends with a Radio Resource Control (RRC) connection reconfiguration message sent by the eNB to the UE. Amongst many pieces of information, it contains neighboring cell identifiers used by the UE to measure the Cell-specific References Signals (CRS).

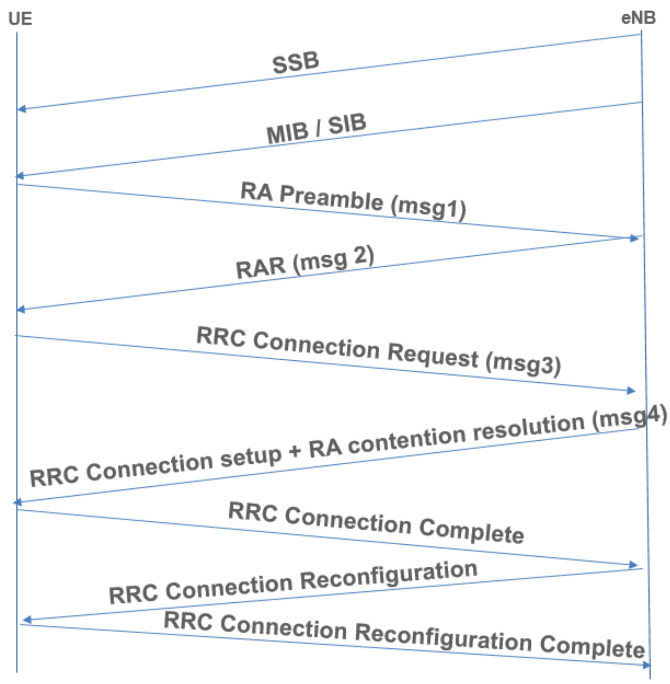


Figure 3.4-2 – Message flow of contention-based RA procedure

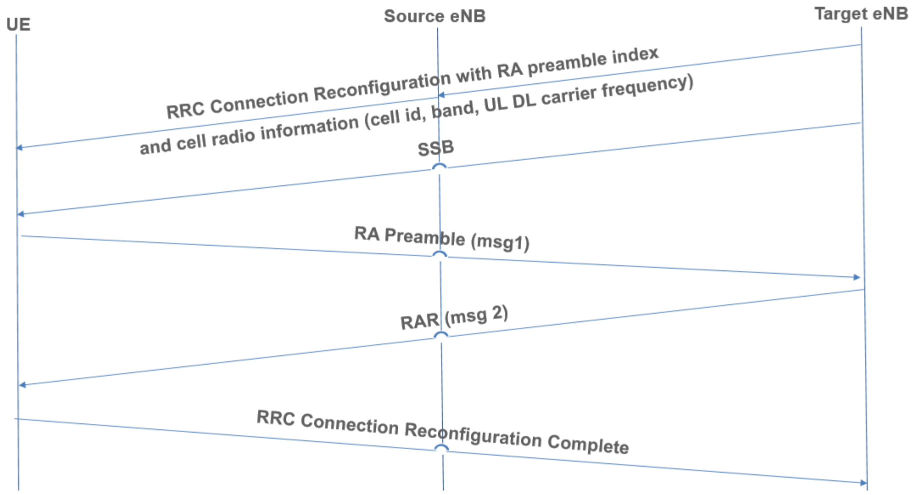


Figure 3.4-3 – Message flow of contention-free RA procedure

Once this procedure is complete, the eNB asks the SPGW to establish a data bearer, then the UE is connected to the CN, and the eNB regularly requests information with Downlink Control Information (DCI) messages. For example, it asks the UE for the status of its uplink buffer, and the UE replies by sending a Buffer Status Report (BSR) message. This exchange is referred to as "DCI to ULSCH traffic" in Figure 3.4-5. The downlink (DL) is transferred from the SPGW to the eNB, which sends it to the UE ("1" and "2" in Figure 3.4-5).

The eNBs are connected to the CN through the S1 interface. The CN is composed of MME and HSS that manage mobility and user connections, SPGW that routes user traffic and user control, and PCRF that manages the plan and billing.

3.4.2.2. Handover

As stated in the motivation, HO is a mechanism that provides service continuity for a UE whose connection needs to be switched from the serving to a chosen neighboring cell.

Handover condition:

Once per subframe (1ms), the UE analyzes the CRS of the serving cell and neighboring cells to estimate their Reference Signal Received Power (RSRP). If the RSRP of the serving cell and neighboring cells verify the input condition during the Time To Trigger (TTT) without verifying the output condition, the UE sends a measurement report to the serving eNB. This report includes the power values and the ID of the reported cell.

The input condition is defined as:

$$Mn + Ofn + Ocn - Hys > Ms + Ofs + Ocs + Off$$

The output condition is defined as:

$$Mn + Ofn + Ocn + Hys < Ms + Ofs + Ocs + Off$$

where

Mn, Ms are measurement (i.e. RSRP in dBm) of the neighboring or the serving cell, respectively, Ofn, Ofs are frequency-specific offset, depending on the frequency of the neighboring cell or the serving cell, respectively (in dB), Ocn, Ocs are cell-specific offset of the neighboring cell or the serving cell, respectively (in dB), Hys is hysteresis parameter (in dB) and Off is an event offset (in dB).

X2 / S1 Handover:

LTE defines two types of HO: X2 HO and S1 HO, which are shown in Figure 3.4-4.

In X2 HO, DL data from the CN to the UE continues to be sent to the source eNB, which forwards it to target eNB via X2 (Data Plane 1 in Figure 3.4-4). Once the UE is connected to the target eNB, a path change is requested by the target eNB to the MME, so the DL traffic is directly forwarded to the target eNB.

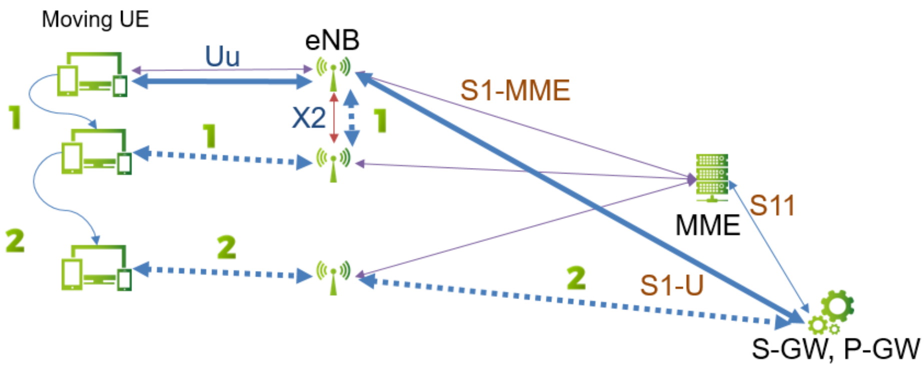


Figure 3.4-4 – Data plane path for LTE end to end architecture during X2 HO (1) and intra-MME S1 HO (2)

S1 HO occurs when the MME serving the source eNB and the target eNB are different, or when the X2 link is unavailable. In such case, there is no path switch. The CN creates a new bearer, and the DL data is buffered to the target eNB via S1 link through the newly created bearer. In both cases, after the completion of HO, the data buffered at the target eNB during HO is sent to the UE before any other data. In this work, we consider X2 HO.

Handover Message Flow:

Figure 3.4-5 details the X2 HO message flow.

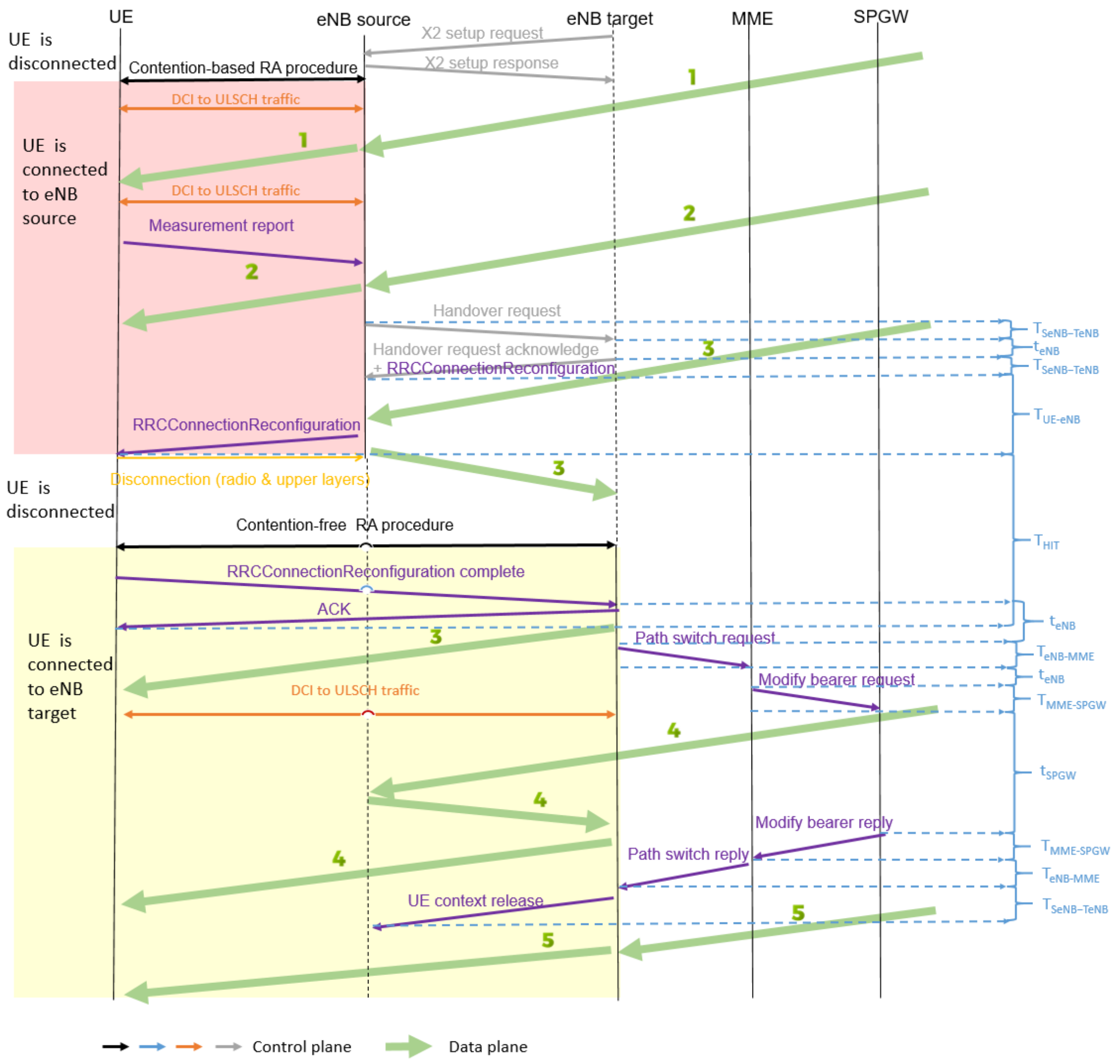


Figure 3.4-5 – Control and data message flow before, during and after an X2 handover

Initially, the X2 link is established between the eNBs and the UE connects to the source eNB. Then, the UE moves away from the source eNB to the target eNB. It meets the HO conditions and sends a measurement report to the serving eNB. The source eNB requests, via the X2 interface, the target eNB to perform the HO with a HO request message. If the target eNB has enough resources (radio, computing, scheduling), it responds with an HO request acknowledgement message. It also sends a RRC connection reconfiguration message, which the source eNB forwards to the UE. This RRC message includes a random access preamble and a mobility control information section. With the presence of this section, the UE knows that it must perform an HO. It reconfigures its layers according to the previous RRC message and disconnects from the source eNB. From this point on, the UE cannot receive or send any information. Since this is an X2 HO, the DL data destined for the UE is still sent by the SPGW to the source eNB, which redirects it to the target eNB ("3" and "4" in Figure 3.4-5) until bearer path change is completed. The UE looks for the synchronisation signal from the target eNB, and then performs a CFRA procedure. As soon as it receives its identifier from the target eNB, it replies with a RRC reconfiguration complete message and the target eNB requests to the SPGW to change the bearer path. When the change is made, the source eNB releases its UE context and no longer receives DL data addressed to the UE: The CN sends DL data directly to the target eNB ("5" in Figure 3.4-5).

Handover Latency Decomposition:

Han et al. [HSC+15] propose the following equation which expresses the total HO time T_{HO}. This equation is composed of terms according to the three HO phases: preparation, execution and completion.

$$T_{HO} = T_{HO\ Prep} + T_{HO\ Exe} + T_{HO\ Comp}$$

The HO preparation time $T_{HO\ Prep}$ is from the time the source eNB receives measurement the report to the time the UE receives the RRC connection reconfiguration message.

$$T_{HO\ Prep} = 2T_{SeNB\text{-}TeNB} + t_{eNB}$$

Where $T_{SeNB\text{-}TeNB}$ is the latency experienced through the X2 link and t_{eNB} is the processing time at the eNB.

The HO execution time $T_{HO\ Exe}$ is between the end of the preparation phase and the moment when the UE receives the RRC connection reconfiguration complete acknowledgement.

$$T_{HO\ Exe} = T_{HIT} + T_{UE\text{-}eNB}$$

Where T_{HIT} is the time between receiving the RRC connection reconfiguration message and receiving the ACK of RRC connection reconfiguration complete (UE receives ACK from eNB) and $T_{UE\text{-}eNB}$ is the time taken to transmit RC connection reconfiguration (eNB to UE).

The HO completion time $T_{HO\ Comp}$ is from the time the target eNB receives RRC connection reconfiguration complete message until the source eNB releases the UE context.

$$T_{HO\ Comp} = 2T_{eNB\text{-}MME} + 2T_{MME\text{-}PGW} + T_{IP\text{-}CAN} + T_{SeNB\text{-}TeNB} + t_{SPGW} + t_{MME} + t_{eNB}$$

Where $T_{eNB\text{-}MME}$ is the latency experienced by the S1-MME link, $T_{MME\text{-}PGW}$ is the latency experienced by the S11 link, $T_{IP\text{-}CAN}$ is the time needed to change the bearer at the CN side (SPGW and PCRF), $T_{SeNB\text{-}TeNB}$ is the latency experienced by the X2 link, t_{SPGW} is the processing time at the SPGW, t_{MME} is the processing time at the MME and t_{eNB} is the processing time at the eNB.

In our testbed, OAI does not include a PCRF. Thus there is no $T_{IP\text{-}CAN}$ processing. Moreover, the target eNB sends the RRC connection reconfiguration complete ACK in parallel with the path change request. This parallelization of procedures simplifies the total HO time equation as follows:

$$T_{HO} = 3T_{SeNB\text{-}TeNB} + 2T_{MME\text{-}SPGW} + 2T_{eNB\text{-}MME} + T_{HIT} + t_{MME} + t_{SPGW} + 2t_{eNB}$$

3.4.3. Numerical results

In this section, we present our experimental setup as shown in Figure 3.4-6.

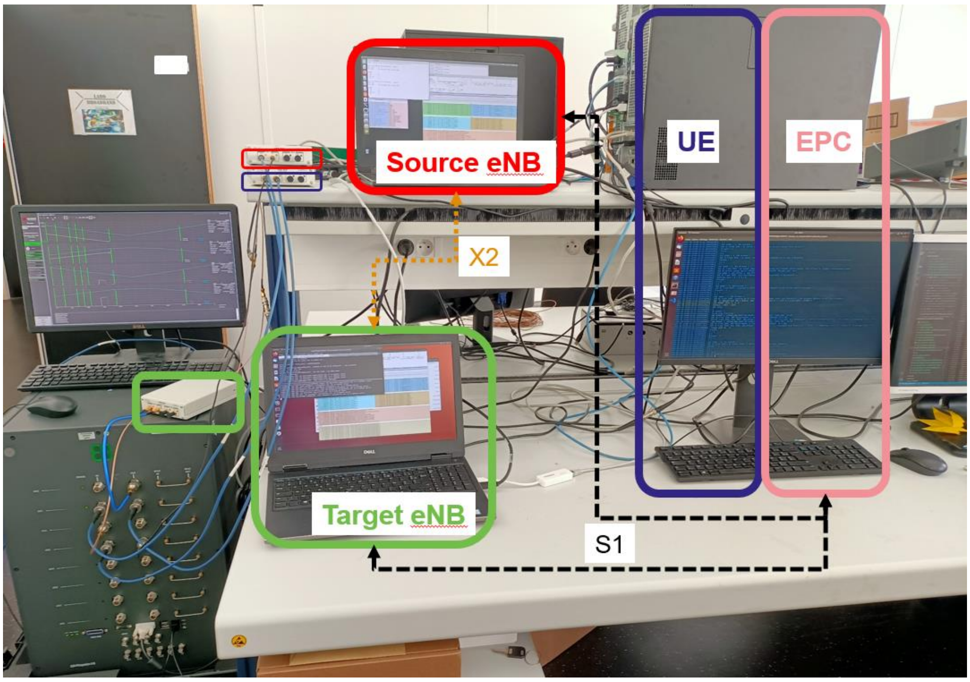


Figure 3.4-6 – OAI-based experimental setup for scenario 2

Our testbed consists of 3 USRPs connected to 3 high-end computers. The target eNB is composed of a USRP x310 in scenario 1, a USRP b210 in scenario 2, connected to a laptop (green rectangles in Figure 3.4-6) while the source eNB is a USRP b210 connected to another laptop (red rectangles). The UE consists of a USRP b210 connected to a desktop machine (blue rectangles). This computer also hosts a virtual machine that runs the OAI EPC (pink rectangle). This configuration can represent a MEC-enabled network architecture, or a private network in a factory where the CN is close to RAN.

Since 2.68GHz belongs to the licensed band 7 in Europe, we use SMA cables instead of antennas to interconnect USRPs. As recommended by Ettus for loopback configurations, in scenario 1 the target eNB and the UE have a static 30dB attenuation at their respective Tx. Table 3.4-1 summarizes the LTE network parameters.

Table 3.4-1 Experimentation parameters

LTE Parameters	**Value**
FDD/TDD	FDD
Downlink Central Frequency	2.68 GHz
Bandwidth	5MHz
Source eNB output power at CF	-60 dBm
Attenuator at Source eNB Tx *	Variable (0-122 dB)
Target eNB output power at CF	-65 dBm
Attenuator at Target eNB Tx*	40 dB
PHICH	1/6
PRACH Configuration Index	0
Max RACH TX	10
RACH Power Ramping Step	4
Handover Parameters	**Value**
Hys	2 dBm
Off = Ofn = Ofs = Ocn = Ocs	0 dB
TTT	40 ms

*: For the experimentation in scenario 1 only. In scenario 2, we define a shadowing profile in the channel emulator to emulate UE mobility.

We have defined two scenarios. Scenario 1 uses a variable attenuator connected to the Tx output of the source eNB. In order to activate the HO condition, we manually increase the DL attenuation of the serving cell in 1 dB steps, which results in a decrease in the RSRP measured by the UE, as depicted in Figure 3.4-7. The lower peak preceding each plateau is due to the attenuation controller. Since the duration of the peaks is between 1 and 7 ms, this has no impact on the overall handover process. Indeed, the RSRP measurement and filtering do not take in account the previous measurements and the duration of the peak is less than TTT (40ms in our setup).

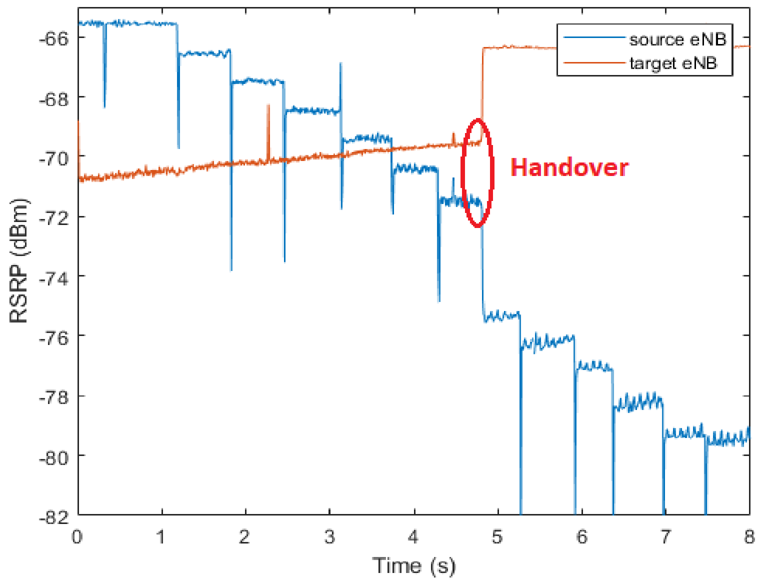


Figure 3.4-7 – RSRP measured over time, post L3 filtering, scenario 1

Scenario 2 uses a Propsim F8 channel emulator in order to have a realistic channel, i.e., the 3GPP Extended Pedestrian A (EPA) channel model, with a 5Hz doppler. The corresponding RSRP measurement is depicted in Figure 3.4-8. Mobility is simulated by a shadowing profile that regulates the attenuation between all transceivers: the signals from the source eNB become weaker and the signals from the target eNB become stronger, which is a more realistic scenario than scenario 1 where the only signal whose strength changes over time is the Tx signal from the source eNB to the UE.

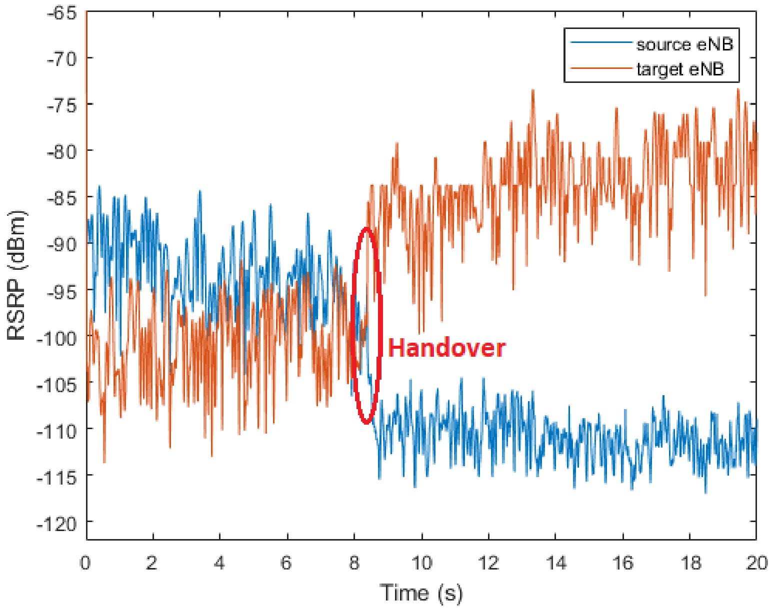


Figure 3.4-8 – RSRP measured over time, post L3 filtering, scenario 2

In Figure 3.4-7, at t=3.73s, the RSRP of the target eNB is greater than the RSRP of the source eNB, but it is not high enough to enter the input condition. At t=4.64s, the HO condition is satisfied and there are no leaving condition for 40ms. Therefore, the UE sends a measurement report to the serving cell. At t=4.76s, the UE receives the HO command, the HO procedure starts and the UE connects to the target eNB. While the UE is performing the HO, it does not measure the RSRP. Measurements after t=4.76s are made after the HO.

Between been well received t=4s and t=5s, we observe a 3dB jump or drop in the measured RSRP of the target eNB and the source eNB respectively. This is due to the fact that the HO procedure induces a frequency change (i.e. an offset in the carrier frequency of the experimental setup, even though the DL frequency is supposed to be the same).

Figure 3.4-9 is a screenshot of a part of the OAI GUI with several lines. The top 3 lines represent the downlink control, its corresponding ACK and NACK respectively; and the bottom 2 lines represent the uplink control and its corresponding ACK respectively. The regions highlighted in red and yellow correspond to the periods when the UE is connected to the source and target eNB, respectively.

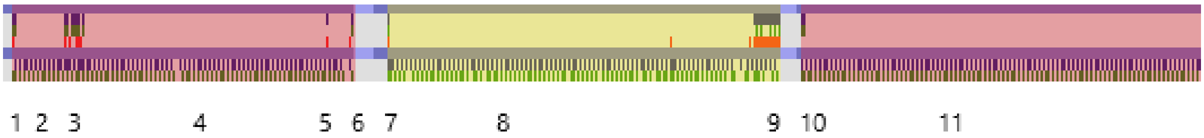


Figure 3.4-9 – OAI GUI capturing two HOs from the UE perspective

The different steps of the HO procedure are:

1. The UE launches its random access to the source eNB (RRC: idle to connected).
2. The UE is connected to the source eNB without CN connection (RRC connected).
3. After NAS exchanges, the UE is connected to the CN (RRC connected).
4. The UE is connected to the source eNB with core connection (RRC connected): Stable UL/DL.
5. The UE sends a measurement report to the source eNB (RRC connected).
6. The UE receives a RRC connection reconfiguration (RRC connected to idle). Start of HIT, start of HO.
7. The RRC connection reconfiguration is completed to the target eNB (RRC idle to connected). End of HIT.
8. The UE is connected to the target eNB (RRC connected).
9. The UE sends a measurement report to the target eNB + retransmissions due to failures + reception of the RRC connection reconfiguration.
10. The RRC connection reconfiguration is complete to the source eNB. End of the second HO.
11. The UE is connected to the source eNB.

Figure 3.4-10 shows the latency decomposition according to the different times defined in the total HO time equation. The duration of the three HO phases is 83 ms, 150ms and 8ms for preparation, execution and completion phases respectively. The end of the execution phase and the beginning of the completion phase are done in parallel, resulting in a gain of 3 ms. Thus the total HO time is 238 ms.

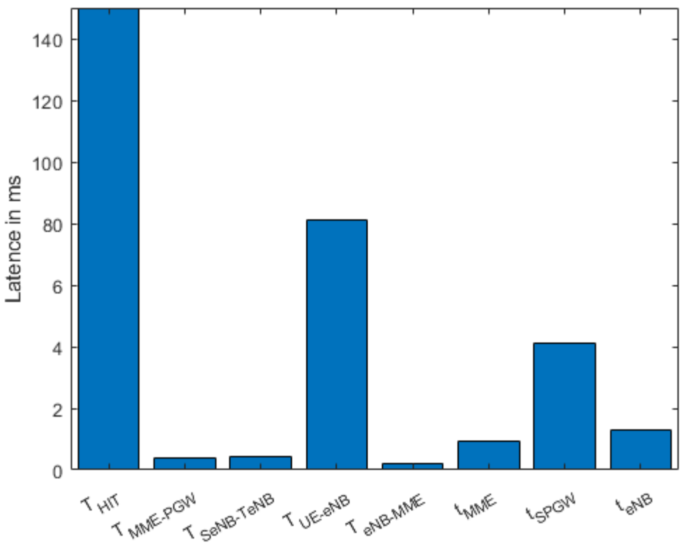


Figure 3.4-10 – Latencies measured

Specifically for the RRC connection reconfiguration time, T_{HIT} includes the duration of CFO estimation and the signal synchronisation (i.e. 124 ms) and the CFRA procedure (i.e. 12ms). We use a single channel UE USRP board, so the CFO cannot be estimated on-the-fly. The UE scans CFO after breaking the connection to the source eNB, resulting in a reasonably large T_{HIT}. The value of 12ms depends on the PRACH configuration index chosen by the eNB. Indeed, after synchronisation, the UE waits for an RA procedure opportunity. We choose the PRACH configuration index 0 which offers an RA opportunity only at subframe 1 of even subframes, i.e. once every 20ms. The UE synchronizes with the eNB by decoding the primary and secondary synchronization signals (PSS/SSS) that are broadcasted in subframe 0 and 5. Thus, if the UE synchronizes to an even-numbered frame at subframe 5, the opportunity to send a preamble will be at subframe 1 of frame N+2, 16ms later.

Moreover, in our experimental setup, there is no PCRF. Since the $T_{IP\text{-}CAN}$ component consists of the processing time in the PCRF and P-GW and we know the exact processing time in the P-GW, which is 4ms, assuming [HSC+15] measurements are consistent with our data, we can estimate the processing time in the PCRF as 20ms-4ms=16ms.

Figure 3.4-11 shows the end-to-end uplink throughput, using the client-server tool iperf for generating UDP traffic. When connected to the CN, the UE receives an IPv4 address. To generate the uplink traffic, we bind the client to the IP address of the UE and the server to a machine routed to UE via CN. The throughput is measured on the server side. We first see that the throughput before handover is about 7Mbps. We can see a drop in throughput at t=2.6s, a complete loss of traffic at t=2.7s and a start of recovery at t=2.8s to finally get back to the cruising speed at t=2.9s. The 0.2ms between the drop in throughput and the recovery corresponds to the duration of the handover.

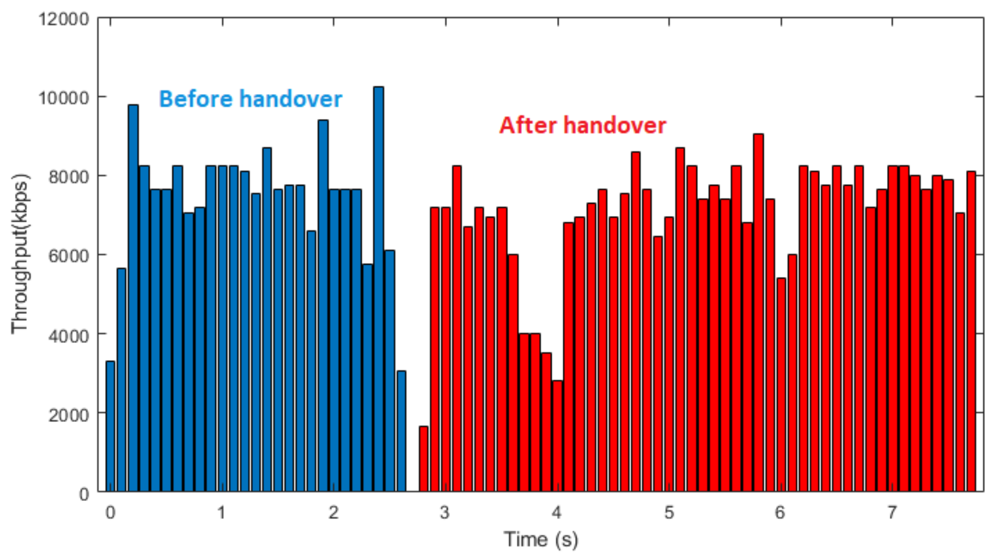


Figure 3.4-11 – End to End uplink throughput measurement

3.4.4. Conclusion

In this study, we present an experimental testbed of the X2 handover procedure using an accessible and reconfigurable software defined radio environment with an end-to-end architecture (i.e. including both the radio access network and the core network). We find that most latencies are of the same order as those found in the state of the art of real experiments. The configurability and openness of the setup, however, brings a tradeoff, namely the latency caused by the reconfiguration of the UE to align its frequency with the target CFO. Last but not least, OAI currently only supports Non-StandAlone (NSA) network with COTS UEs. In future work, we plan to integrate the X2 handover procedure into 5G to enable 5G NSA using an OAI UE instead of a COTS UE.

3.5. Resource Scheduling Optimization for Ultra-Reliable Low-Latency Communications

3.5.1. Motivation and related work

In dynamic channel conditions, latency of a packet depends on the number of retransmissions necessary to correctly receive it. In 4G and 5G, retransmissions are scheduled according to Hybrid Automatic Repeat reQuest (HARQ) procedure. Each transmission follows an HARQ process. If the transmission is not acknowledged, i.e. if the sender doesn't receive any feedback or if the received feedback is a NACK, retransmissions are triggered. Each retransmission follows the former rule, until the sender receives an ACK feedback or until I reaches a predefined maximum of retransmissions. This sequential procedure improves reliability at the expense of the latency. Indeed, the greater the number of retransmissions, the higher the probability of a successful transmission; however the longer the procedure takes to complete.

To improve the latency, it is thus necessary to adapt the procedure depending on the conditions, of the channel but also of the current going traffic. In the literature, adaptation of the HARQ strategy is usually achieved by adapting the modulation and coding scheme [PDN14], the transmission power [JBSL16] and the maximum number of retransmissions [MAA16] but rarely at the scheduler level. A K-repetition scheme [LSK21] and a proactive scheme with early termination [LDE+21] have been proposed, allowing for a number of redundant retransmissions upon receipt of the acknowledgment by the sender. By doing so, one can opportunistically decode the packet at the receiver in a shorter time at the expense of inefficient resource usage [JAB+17]. However, the adaptation of HARQ strategies at the scheduling level in a rapidly changing environment is limited in current research.

The work done by Lam Ngoc Dinh et al [DLMC22] attempts to fill this gap by exploring the parallelisation of HARQ procedure with a resource scheduling optimization algorithm. In this section, we present an evolution of this work. Next, we implement in our 5G OpenAirInterface testbed the algorithm introduced in [DLMC22] and its evolution, in order to bring a proof of their feasibility under real time restrictions, and evaluate their performances in experimentation. For the sake of simplicity, we focus only on Downlink transmission.

3.5.2. HARQ Procedure

3.5.2.1. **General procedure : Reactive HARQ**

The HARQ procedure is standardized by the 3GPP consortium, and involves control transmission between the sender and the receiver. The procedure flow is depicted in Figure 3.5-1. It aims to improve the reliability of the communication by triggering retransmission of a packet that is assumed not to have been well received by the receiver. Supposedly, because a base station assumes that a retransmission is needed when either:

- gNB received a NACK feedback; or
- gNB does not receive any feedback.

In the second case, it is possible that the UE well received the packet. Feedback might have been lost in the transmission. In such case, because gNB has no mean to know the status of the packet at the received side, it assumes that the packet has not been well received.

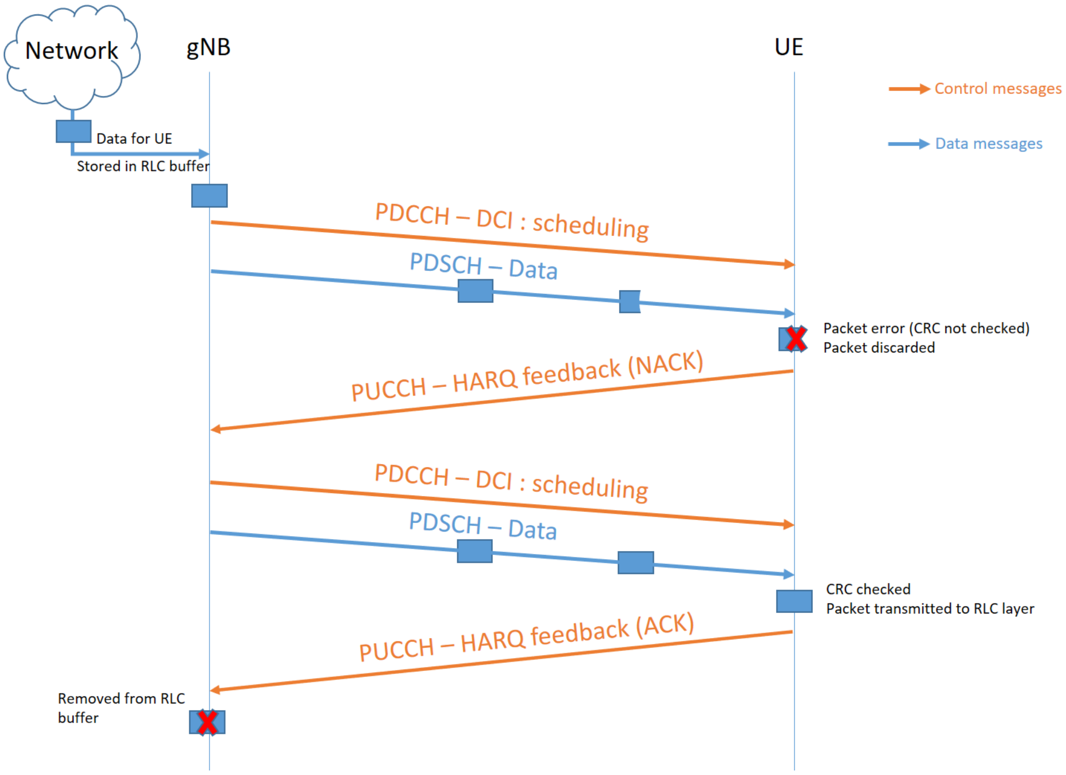


Figure 3.5-1 – HARQ Procedure Flow : Reactive

We take a simple example scenario: some server outside of local network has a packet to send to the UE. The packet reaches the gNB, goes through the PDCP layer and ends up in the Tx buffer of the RLC layer. It waits until an HARQ process picks it up for an initial transmission. Once picked, it is removed from RLC Tx buffer, and is duplicated to the RLC reTx buffer. It then trigger the scheduling of transmission of multiple information:

- the transmission of the actual data
- the transmission of the feedback
- the transmission of the DCI that contains the above scheduling information

Let's say that the UE fails to decode the packet (CRC not checked, SNR too low…), it sends a NACK to the gNB to ask for a retransmission. The HARQ process that is in charge of current packet will then trigger again a new scheduling of the above mentioned transmissions. This loop is done until either gNB receives an ACK, either the maximum number of retransmissions is reached. The maximum number of retransmissions is not standardized, it is up to the manufacturer / implementer to choose a number that fits the constraint of the system.

As shown in Figure 3.5-1, the reactive scheme induces a latency that grows linearly with the number of retransmissions needed. However as a single retransmission is scheduled each time, we allocate exactly the resources needed for reaching a given reliability. In the context of URLLC however, we want to give the latency a higher weight than that of the resource allocation efficiency.

3.5.2.2. K-repetition HARQ

Research in retransmission schemes introduced the K-repetition scheme. As shown in Figure 3.5-2, instead of waiting for a feedback before making the decision to trigger or not a retransmission, the gNB retransmits in parallel K times the packet. It will then wait for all of the feedback before making the decision. If it receives at least one ACK or if the maximum number of retransmissions is reached, the HARQ procedure for current packet is terminated. Otherwise it will send again in parallel K times the packet.

This scheme reduces the latency of the HARQ procedure completion, but it increases the waste in resource allocation. Also the K is given arbitrarily and does not adapt to the system current state, excluding thus any adaptation to the channel state. We have a high probability to eventually face situations of over-allocation or increasing latency. It works best in a well-known environment with little change over time.

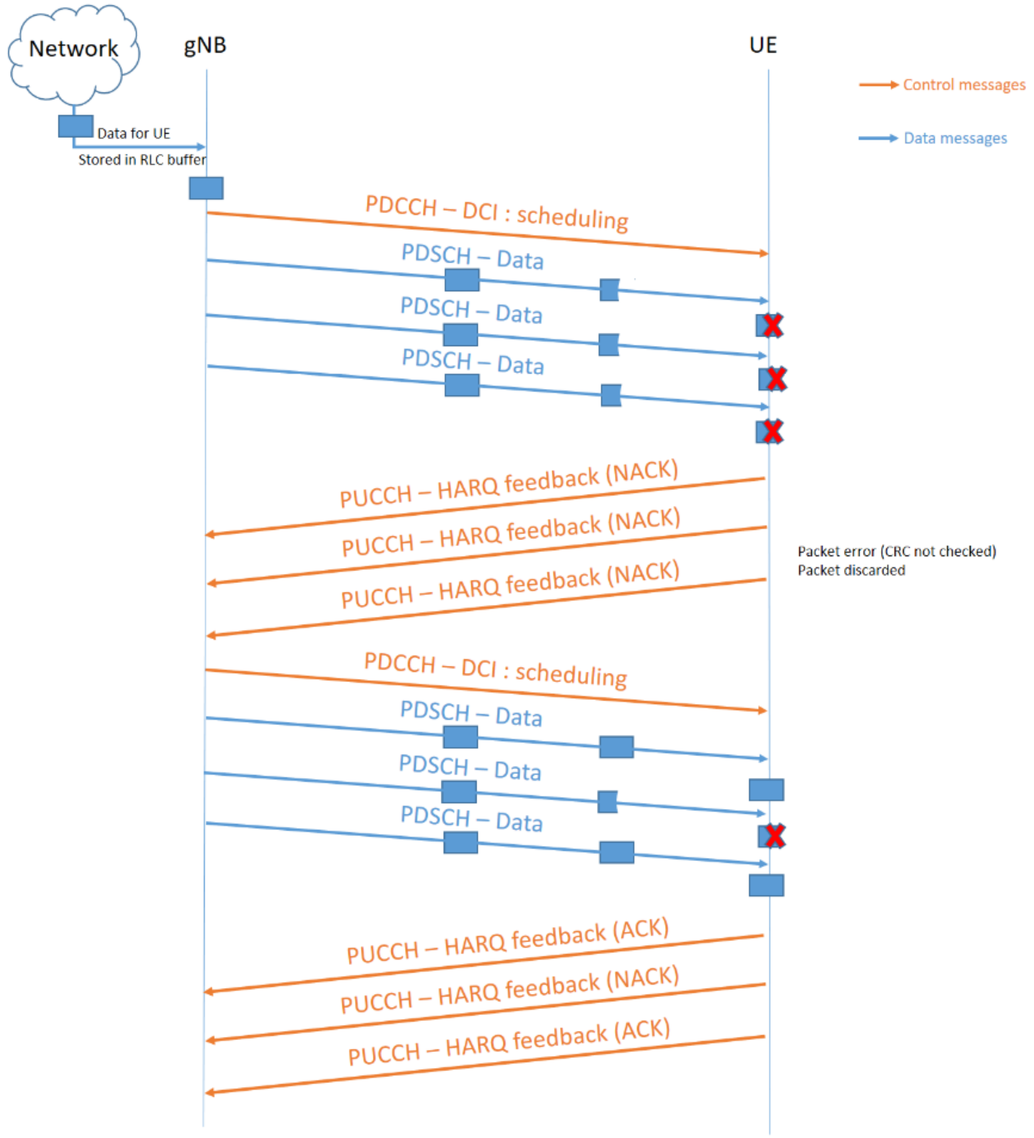


Figure 3.5-2 – HARQ Procedure Flow : K-repetition

3.5.2.3. Dynamic Resource Scheduling Optimizer with Lyapunov

To overcome the drawbacks of a statically assigned K, we introduce an algorithm of dynamic resource scheduling optimization with Lyapunov optimization.

3.5.2.3.1. System Model

In this section, we describe the system model making the trade-off between the latency, the reliability and the resource efficiency. A series of actions $a_j \in \{a_0, \ldots, a_{max}\}$ is made at the corresponding action slot t. We can define two queues: The arrival rate queue $Q_1(t)$ is the RLC transmission buffer and contains the application packets. After completing the scheduler operations at MAC layer, the gNB prepares a Transport Block (TB) whose data is extracted from $Q_1(t)$ and sends it over the air. The scheduling rate queue $Q_2(t)$ keeps a copy of this TB and takes into account the ongoing scheduling processes that are not yet decoded at the UE side. Due to the dynamic nature of not only the traffic but also the channel behaviour, the lengths of $Q_1(t)$ and $Q_2(t)$ can be considered as random variables. The state of $Q_1(t)$ and $Q_2(t)$ demonstrated a two-stage queuing system whose length should be minimized. The queuing dynamic is defined as follows:

$$Q_1(t+1) = \max\{Q_1(t) - \alpha_a \times TB_{a_0}, 0\} + A_1(t)$$

$$Q_2(t+1) = \max\left\{Q_2(t) - (1 - \alpha_a) \times 1_{TB} \times TB_{a_j}, 0\right\} + A_2(t)$$

where $Q_1(t+1)$ are the backlogs of the queue i at the action slot $t+1$. $A_1(t)$ represents the total amount of high layer packets that arrive Q_1 at time t. During this action slot, an amount of TB_{a_j} will be served. The indicator function 1_{TB}, in the second equation, is equal to 1 if the scheduling process of TB is successful and is 0, otherwise. If the first transmission of TB_{a_0} is a failure, $A_2(t) = TB_a$ will be added to Q_2, otherwise $A_2(t) = 0$ as the scheduling process of TB_{a_0} is ending. In order to control which queue will be served, we introduced the control variable α_a (1 and 0 mean serving $Q_1(t)$ and $Q_2(t)$ respectively). Knowing that ongoing processes have a higher priority, $\alpha_a = 0\ Q_2(t) > 0$.

3.5.2.3.2. Dynamic Resource Scheduling in HARQ Procedure

In this section, we apply the system model to HARQ procedure. Figure 3.5-1 first illustrates the classic HARQ procedure (i.e. send-wait-react mode).

A delay L_{12} is introduced to demonstrate the TB preparation time from the gNB scheduler to the antenna. Then, a feedback will be encoded within an Uplink Control Information (UCI) message and sent back to the gNB after $T_{fb} = K_1$ slot(s), thus illustrating the processing time at the UE. In 5G NR standard, this processing time reflects a delay between the reception of the UL grant in the DL and the transmission of the corresponding UL data. Afterwards, the gNB has the information about the corrupted HARQ process on the UE side and decides to retransmit the erroneous TB after L_{12} slots.

This process continues until the corrupted TB is successfully decoded by the UE or the maximum number of retransmissions R_{max} is reached. By doing this, the resources are perfectly utilized, but the latency could be unacceptable for URLLC communications. Instead of limiting the maximum number of allowed RTXs R_{max} for the scheduling process, our dynamic resource scheduling is restricted in terms of maximal possible actions a_{max}. Each action $a_j \in \{a_0, \dots, a_{max}\}$ can allocate r_{a_j} proactive RTXs, between r_{min} and r_{max}, and a_0 corresponds to the first transmission.

The decision maker we designed, dynamically chooses the number of actions a_j and their intensities (i.e, r_{a_j}) to reduce latency and improve resource efficiency and reliability. With respect to the resources allocated for proactive RTXs of a TB_n, the decision maker selects an element-wise positive resource allocation vector $\left(r_{n,a_0}, r_{n,a_1}, \dots, r_{n,a_{max}}\right)$. If r_{n,a_j} proactive RTXs are allocated by action a_j , the risk (i.e $\zeta(a_j)$) is expressed as follows:

$$\zeta(a_j) = \mathbb{P}\left[\left(SINR_{tb_n}^{\sum_{k=0}^{j} r_{n,a_k}} \leq SINR^t\right) \mid SINR_{tb_n}^{\sum_{k=0}^{j-1} r_{n,a_k}}\right]$$

where $SINR_{tb_n}^{\sum_{k=0}^{j-1} r_{n,a_k}}$, $SINR_{tb_n}^{\sum_{k=0}^{j} r_{n,a_k}}$ are respectively the SINR of TB_n at previous$\left(a_{j-1}\right)$ and current $\left(a_j\right)$ action. $SINR^t$ is the target SINR to decode TB_n.

Our proposed procedure dynamically adapts the resource scheduling to the traffic arrival in the network layer, the queue behaviours in the data link layer and the risk that the applied decision causes loss. It also automatically adapts the maximum number of RTXs to the channel conditions. Finally, to reduce the control overhead due to multiple feedbacks to the transmitter, we grouped their feedbacks into a single feedback that represents the current proactive retransmission status

3.5.2.3.3. Problem Formulation

The objective is to optimally select r_{a_j} based on various factors, such as the current status of the $Q_1(t)$, $Q_2(t)$, the current action index a_j and the risk that the applied decision causes loss. The main reliability constraint is to reduce the risk of the last action $\zeta(a_{max})$ below a predefined value ζo. However, the constraint associated with poor decision making must be defined for each upcoming action, not just for the last action. We define the risk for the current action $\zeta(a_j)$. In this case, the procedure has to retrigger other actions later, which consumes not only time and resources but also the reliability of the communication, when we are close to the maximum number of actions allowed. The index of the current action (i.e a_j) is thus very important. Clearly, the higher j is, the greater the sensitivity of TB loss will be if a wrong decision is applied, and the earlier the action (i.e low j) is, the higher the total number of RTXs can be. We define an objective function f_{obj} as the weighed sum of average number of resources allocated to each TB and the current risk, as follows:

$$f_{obj} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \sum_{a_0}^{a_{max}} r_{n,a_j} \times 1_{a_j} + \alpha \times f(a_j) \times \zeta(a_j)$$

where the indicator function 1_{a_j} is equal to 0 if the action a_j is successful (i.e. $\zeta(a_j) < \zeta_0$) and is 1, otherwise. $\alpha \geq 0$ is a constant value trading off risk and resource allocation. A higher value of α implies greater importance of risk minimization over the number of resources allocated (i.e. reliability over resource efficiency). The function $f(a_j)$ increases with the action index a_j. In our study, we consider $f(a_j) = j$.

Thus, our optimization problem $\mathcal{P}_1$ is to minimize the objective function f_{obj} subject to several constraints:

$$\min_{\{r_{n,a_j}\}\, n,a_j} f_{obj} \quad (\mathcal{P}_1)$$

$$\text{s.t.} \lim_{t\to\infty} \frac{\mathbb{E}\{Q_i(t)\}}{t} = 0, \forall i \in \{1,2\}; \qquad (\mathcal{C}_{1,2})$$

$$r_{min} \times 1_{a_j} \leq r_{n,a_j} \leq r_{max} \times 1_{a_j}, \forall a_j \leq a_{max} \; (\mathcal{C}_3)$$

$(\mathcal{C}_{1,2})$ concern the stability constraint of the queues $Q_{1,2}(t)$. $(\mathcal{C}_3)$ limits the number of decisions into a_{max} actions and constrains the maximal number of proactive RTXs at action a_j to r_{max}.

Our dynamic decision maker algorithm is based on Lyapunov's optimization tools to solve the optimization problem $(\mathcal{P}_1)$. We define the current state in the slot t as $\Theta(t) = (Q_1(t), Q_2(t)))$ and the Lyapunov function as follows:

$$L(\Theta(t)) = \frac{1}{2}[Q_1^2(t) + Q_2^2(t)]$$

Next, we define the one-slot conditional Lyapunov drift $\Delta(\Theta(t))$ representing the expected change of the Lyapunov function over a slot as follows:

$$\Delta(\Theta(t)) = \mathbb{E}\{L(\Theta(t+1)) - L(\Theta(t)) \mid \Theta(t)\}$$

By minimizing both $\Delta(\Theta(t))$ and f_{obj}, we can solve the problem $(\mathcal{P}_1)$ because the queues are stable in terms of average rate and the objective function is minimized . However, according to [NEE10], a performance-delay trade-off between these dual objective optimizations can be parameterized by a constant V. By setting a large positive value to V, the control algorithm will favor minimizing the objective function f_{obj} over the stability of the average rate queues. Our objective is then to minimize the following Lyapunov-drift-plus-penalty function:

$$g(t) = \Delta(\Theta(t)) + V.\mathbb{E}\{f_{obj} \mid \Theta(t)\}$$

As defined in [13], the upper bound, $\gamma(t)$, can be derived for any action, any possible value of $\Theta(t)$ and any parameter $V > 0$ as follows:

$$\gamma(t) = B + V.\mathbb{E}\{f_{obj} \mid \Theta(t)\} + \sum_{i=1}^{2} Q_i(t).\mathbb{E}\{A_i(t) - b_i(t) \mid \Theta(t)\}$$

where B is a constant that satisfies:

$$B \geq \frac{1}{2}\sum_{i=1}^{2} \mathbb{E}\{A_i^2(t) - b_i^2(t) \mid \Theta(t)\} - \sum_{i=1}^{2} \mathbb{E}\{A_i(t).\min\{Q_i(t), b_i(t)\} \mid \Theta(t)\}$$

Through the opportunistic minimization framework of a conditional expectation [NEE10], by minimizing $\gamma(t)$, the upper bound of the dual objective optimization, we can guarantee that the optimization problem $(\mathcal{P}_1)$ will be satisfied.

3.5.3. Implementation and Experimentation with OpenAirInterface

OpenAirInterface repository is made up of many branches. "master" is the main branch, with releases of tested and validated functionalities. "develop" is the main branch for development. Then other branches are dedicated to the implementation of specific functionalities. Once the functionality has been validated after extensive testing, it is merged into the develop branch. Currently, master branch does not include 5G. It is in a development stage in the develop branch. HARQ implementation in the develop branch of OpenAirInterface includes only the reactive HARQ. We implemented the K-repetition and the Dynamic HARQ schemes in OAI. The core of the work is mainly at the MAC layer, where the scheduler is implemented, but also at the PHY and RLC layers to ensure that the allocated resources are used by the PHY layer, and to take into account the latency of a packet waiting in RLC Tx buffer before an HARQ process retrieves it.

Figure 3.5-3 show the OAI experimentation testbed made up of the following components:

- Two powerful computers
  - 2x14 CPU, 64Gb RAM
- Two SDR cards
  - USRP b210
  - USB 3.0 connection to PC
- Two variable attenuators
  - DL and UL attenuation
- SMA cables, since we are working at 3.61GHz and this frequency belongs to the 5G licensed band.

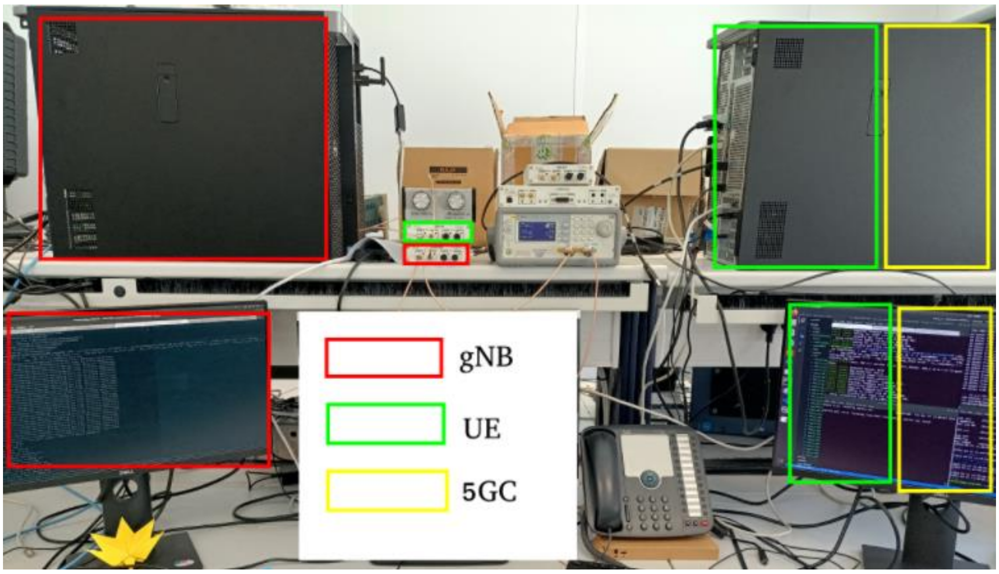


Figure 3.5-3 OAI experimentation testbed

Both computers have OAI installed and synchronized to the same commit from develop branch. The computer on the left side is used to run an instance of an OAI gNB. The computed on the right side is used to run an instance of an OAI UE alongside with the Core Network. We consider that in a situation with few UE, the UE and Core Network are the parts that have less processing time needed, so we gathered both into once computer and let the processing time-hungry gNB make full use of a two-CPU machine.

5G allows many configurations: different numerologies, different frame structures for TDD, etc. The configuration of our 5G testbed is the following:

- Numerology 1: frame slot of 0.5ms
- 50MHz of bandwidth
- Center frequency 3.61GHz
- TDD : 6 slots DL, 1 slot flexible, 3 slots UL
- MCS 3
- Downlink traffic type: 64 bytes ping every 0.05s
- $K_1 = 6$ slots

In our experimental scenario, the variable attenuators are used to experiment several channel conditions between the UE and the gNB. Packets generated are PING of 64 bytes every 50ms. The current status of 5G implementation coupled with the limited capabilities of the USRP do not offer the freedom we would have wanted for experimentations. First, only the TDD mode of 5G is available, which means that there are non-compressible latencies for the HARQ procedure. There is the k0, that represents the delay between the DCI and the PDSCH announced by the DCI; and there is k1 that represents the delay between the PDSCH and the corresponding HARQ feedback.

In our performance evaluation, we have implemented and compared several HARQ schemes: **Classic HARQ** refers to the wait-NACK-before-retransmit scheme, **Parallel HARQ** (from 2 to 5 parallel retransmissions), **Proactive HARQ** as defined in [DLMC22] and our proposed **Dynamic HARQ**. In order to find the suitable values of V (Proactive and Dynamic) and α (Dynamic), we run with different values of V and α and select the value for which a given metric is maximized / minimized.

3.5.4. Numerical results

First, we have to parameter the proactive scheme. Figure 3.5-4 shows the experimentations of the proactive scheme with different values of V.

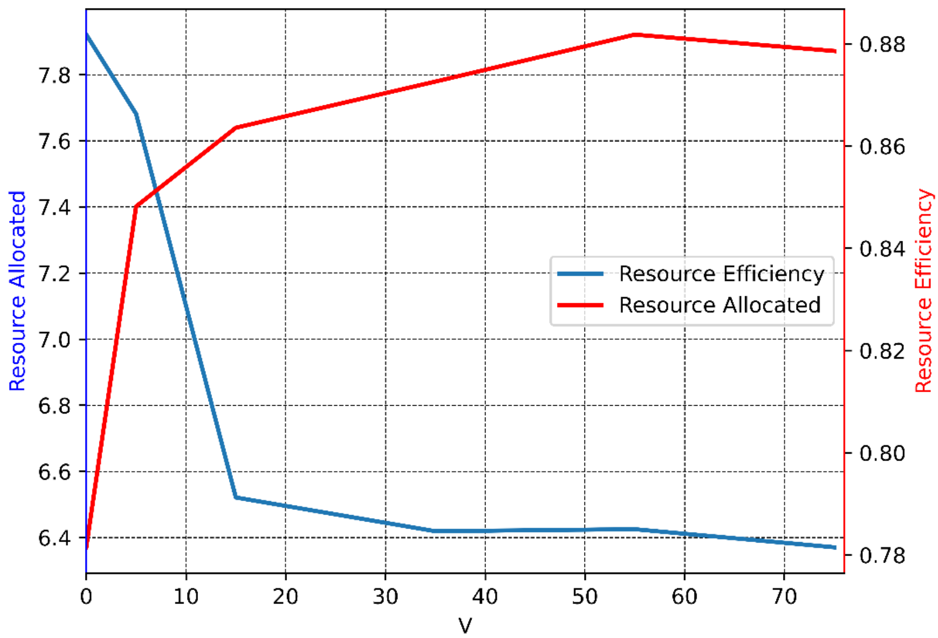


Figure 3.5-4 – Resource Allocation and Resource Allocation Efficiency depending on V

We have to choose a value of V that maximizes the resource allocation efficiency. According to Figure 3.5-4, we set V to 55.

Figure 3.5-5 compares the proactive algorithm with Classic HARQ and Parallel HARQ.

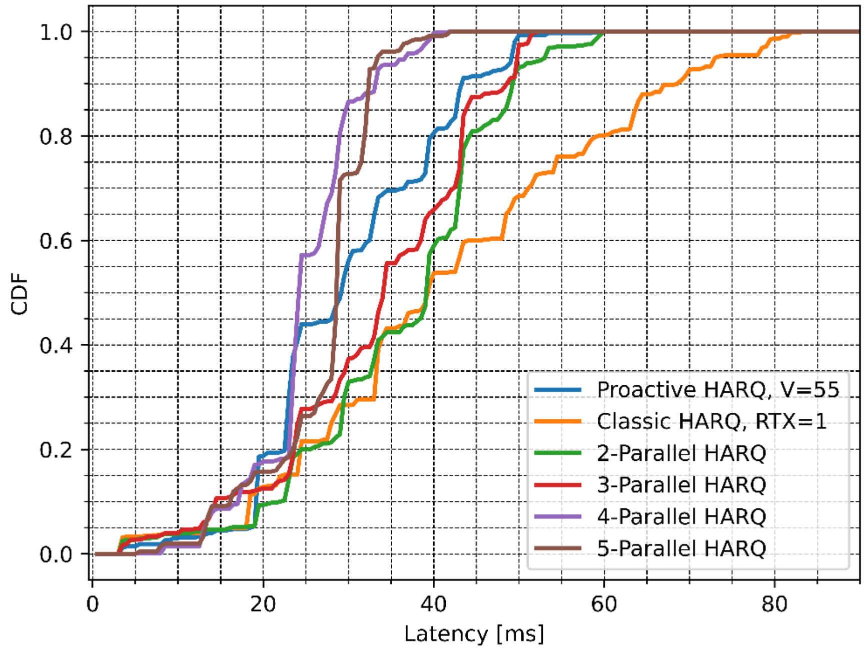


Figure 3.5-5 – Cumulative Distribution Function of Latency completion of HARQ processes for state of the art HARQ schemes

We can see first that Parallel 4 and 5 are very close to each other, which means that in the current testbed with the current channel state, allocating 4 or 5 resources does not improve by factors the performance latency wise: for both, 100% of the packets are located under $42.5ms$. On the other hand, we can see that Classic HARQ gives the worst results with 100% of the packet being under $72.5ms$. Parallel 2 is better with 100% of the packet under $60ms$, Parallel 3 $52.5ms$. Our algorithm, that maximizes the resource allocation, has 100% under $52.5ms$

If we focus on the CDF at 60%, we have the following:

- Classic: $47.5ms$
- Parallel 2: $42.5ms$
- Parallel 3: $37.5ms$
- Parallel 4: $27.5ms$
- Parallel 5: $30ms$
- Proactive [DLMC22], $V = 55$: $32.5ms$

Next, we compare with the Dynamic HARQ implementation. We slightly changed the implementation of the Proactive version: we removed an inner division to artificially increase the value of V. Thus the values of V are not comparable between previous and next results.

In Table 3.5-1, we show the BLER, latency, resource usage and resource usage efficiency depending on the values of V (columns) and α (lines). We highlight three spots.

Table 3.5-1 Numerical results of BLER, Latency, Resource Allocation, Resource Allocation Efficiency depending on α and V

BLER	$V = 0{,}001$	$V = 0{,}5$	$V = 1$	$V = 5$	$V = 10$
$\alpha = 0$	2,23048327	9,3117409	22,163908	27,478754	40,437788
$\alpha = 1$	2,53560264	5,2167783	13,411765	15,201192	21,046611
$\alpha = 10$	2,34246123	2,9767041	3,2193827	4,1683694	4,1200407
$\alpha = 100$	2,27120908	2,4323084	1,6408814	1,9742883	2,1047228
LATENCY	$V = 0{,}001$	$V = 0{,}5$	$V = 1$	$V = 5$	$V = 10$
$\alpha = 0$	19,5289303	25,370268	24,19107	20,536601	21,314922
$\alpha = 1$	22,9312745	22,356983	22,705722	24,841549	26,204719
$\alpha = 10$	20,8545892	20,941006	20,595434	22,084314	20,492283
$\alpha = 100$	22,4786868	21,196891	20,47451	18,221003	18,563746
RESOURCE USAGE	$V = 0{,}001$	$V = 0{,}5$	$V = 1$	$V = 5$	$V = 10$
$\alpha = 0$	7,95089456	7,5285616	6,1858775	4,3751634	3,6472868
$\alpha = 1$	8,92859693	6,4655493	6,0027248	5,1883803	5,553539
$\alpha = 10$	8,11859746	7,0828139	6,5372468	6,7738562	7,0766365
$\alpha = 100$	8,7872121	8,3122939	7,7267974	7,1397223	7,2345226
RESOURCE ALLOCATION EFFICIENCY	$V = 0{,}001$	$V = 0{,}5$	$V = 1$	$V = 5$	$V = 10$
$\alpha = 0$	0,76157418	0,8491635	0,8719154	0,9020018	0,9543039
$\alpha = 1$	0,78419769	0,8316244	0,851566	0,8941296	0,9020261
$\alpha = 10$	0,76627006	0,7922429	0,8130547	0,8214975	0,795217
$\alpha = 100$	0,77540881	0,7732192	0,7611233	0,7407012	0,7296396

First, in purple, with extremely low V or extremely high α the algorithm allocates a high amount of resources. As we are over allocating in current scenario, the resource usage is the highest, from 7 to 9; resulting in the lowest resource efficiency – that is the ratio of resources needed over resources allocated – of around 0.72 to 0.78, which means 28 to 22% of the allocated resources are wasted. However, in such configuration, the BLER is at its lowest: around 2%.

Second, in orange, with $\alpha = 0$ so we do not consider the reliability, and with an extremely high V. In such configuration, we are close to the standard HARQ that would allocate 1 retransmission per cluster. Without surprise, the resource usage is at its lowest, at 3.6; the resource efficiency at its highest, at 0.95; and the BLER is at its lowest with 40% HARQ error.

Finally, in blue, with $\alpha = 1$ and $V = 5$, we have a BLER that is around that of what we see in scenarios with statically allocated 2 and 5 retransmissions. In such case, we see a mean resource usage of 5 and a resource efficiency of 0.89.

Figure 3.5-6 and Figure 3.5-7 show the tradeoffs between resource allocation, resource efficiency and reliability. Figure 3.5-6 shows the relation between resource allocation and resource allocation efficiency with different values of α and V. We can see clearly that the higher the α, the more we allocate, the less efficient we are.

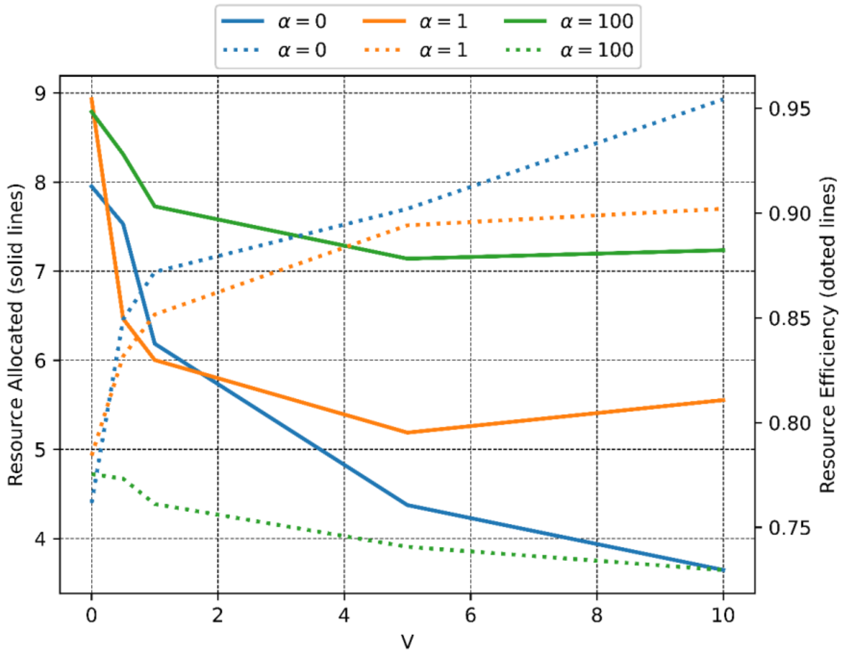


Figure 3.5-6 – Resource allocation and resource allocation efficiency depending on V value for different values of α

Figure 3.5-7 shows the relation between the reliability in terms of BLER and resource allocation efficiency with different values of α and V.

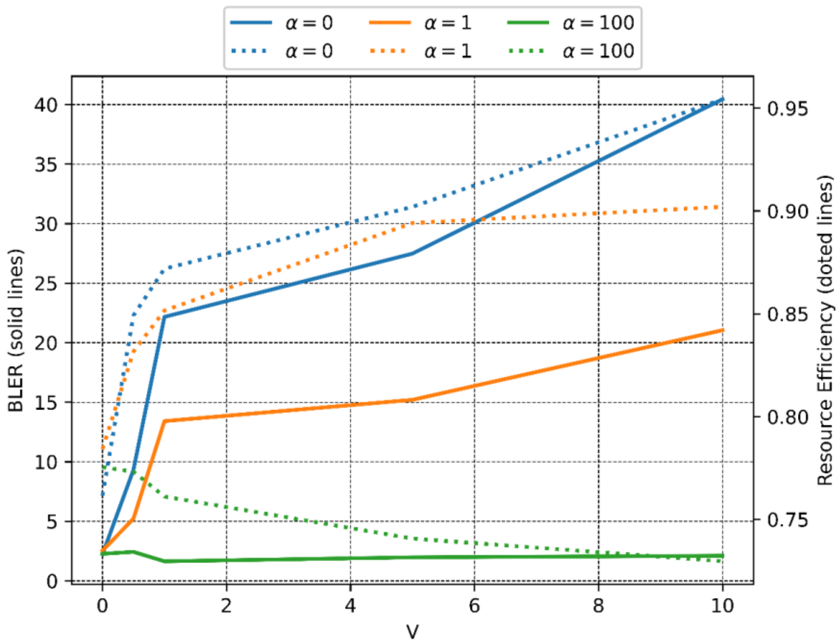


Figure 3.5-7 – Reliability and resource allocation efficiency depending on V value for different values of α

Figure 3.5-8 shows the CDF of latencies of HARQ procedure completion for the following schemes:

- Classic HARQ with wait-NACK-before-retransmit scheme
- 2 and 5-Parallel, that is respectively 2 and 5 parallel retransmissions at each cluster
- Proactive HARQ according to [DLMC22] with $V = 0.06$

- Our Dynamic HARQ with $V = 5$ and $\alpha = 100$ making a reliability-latency-efficiency tradeoff.

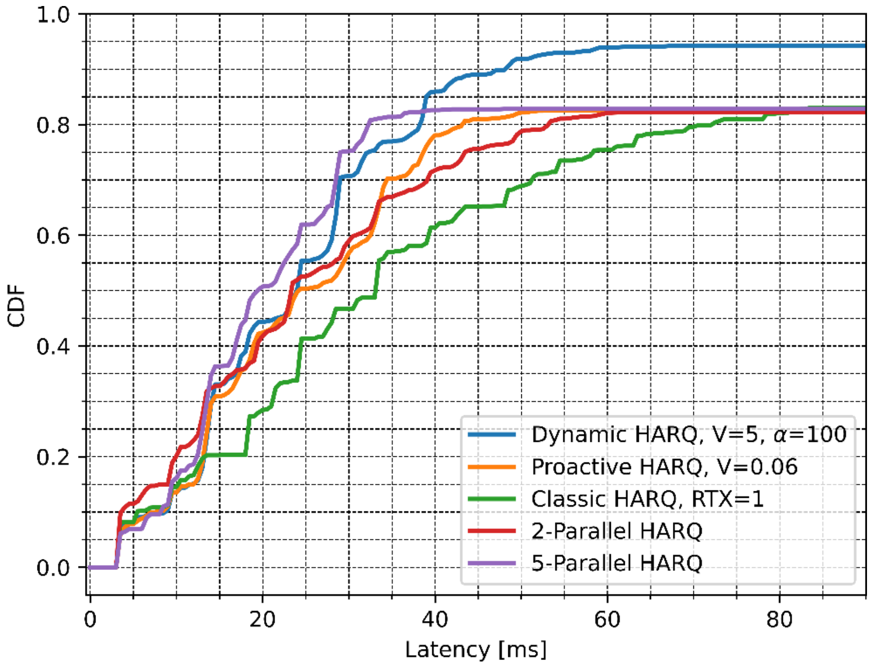


Figure 3.5-8 – Cumulative Distribution Function of Latency completion of HARQ processes for different HARQ schemes

Regarding the resource allocation efficiency, we have the following results:

- Classic: 100%
- Parallel 2: 89%
- Parallel 5: 72%
- Proactive [DLMC22], $V = 0.06$: 82%
- Dynamic, $V = 5$ and $\alpha = 100$: 74%

We can see that in terms of reliability, our algorithm outperforms the four other algorithms because when it is necessary our algorithm allocates more retransmissions. Regarding the latency, it performs less than the 5-Parallel algorithm. However, 5-Parallel algorithm performs less in terms of resource allocation efficiency. Indeed, it is of 72%, and the Dynamic algorithm with $V = 5$ and $\alpha = 100$ has a slightly better efficiency of 74%. For a BLER similar to Parallel 5 BLER, that is with $V = 0.5$ and $\alpha = 0$ (9% BLER) or with $V = 1$ and $\alpha = 1$ (13% BLER), the efficiency is increased to 85%, that is 13% more than Parallel 5 and 3% more than the Proactive with $V = 0.06$.

Finally, we extend the HARQ latency to RLC latency. RLC layer has three different modes:

- Transparent Mode (TM): RLC gives packet to lower MAC layer as they come from the upper layers without any concatenation. Packets are deleted from RLC buffer when they are picked by HARQ processes.
- Unacknowledged Mode (UM): RLC does concatenate upper layer packets when needed before giving them to HARQ processes. In other words, a single HARQ process can handle multiple RLC packets at once. RLC packets are still deleted from RLC buffer when they are picked by an HARQ process.
- Acknowledged mode (AM): RLC does concatenate packets, and it keeps the packet in its buffer until RLC module of receiver acknowledges the well reception (integrity check and other means) of the packet.

Signaling Radio Bearer (SRB) and Data Radio Bearer (DRB) may have different modes within the same RLC layer. In OpenAirInterface, all radio bearers are using AM RLC.

Figure 3.5-9 shows the CDF of the latency at the RLC layer. First we note that the values of the latencies are higher than previously. Indeed, HARQ processes sometimes fail to transmit a packet. As it is AM, the packet remains in RLC until it is well transmitted, thus it may need few HARQ processes to well transmit a packet, and thus the RLC latency is greater than MAC latency by factors. Finally we also note that the RLC latency in the case of Dynamic algorithm is lower than others by factors (x axis is log scale). This is thanks to the high reliability ensured by the algorithm. If less HARQ fail, then packet spend less time in RLC layer.

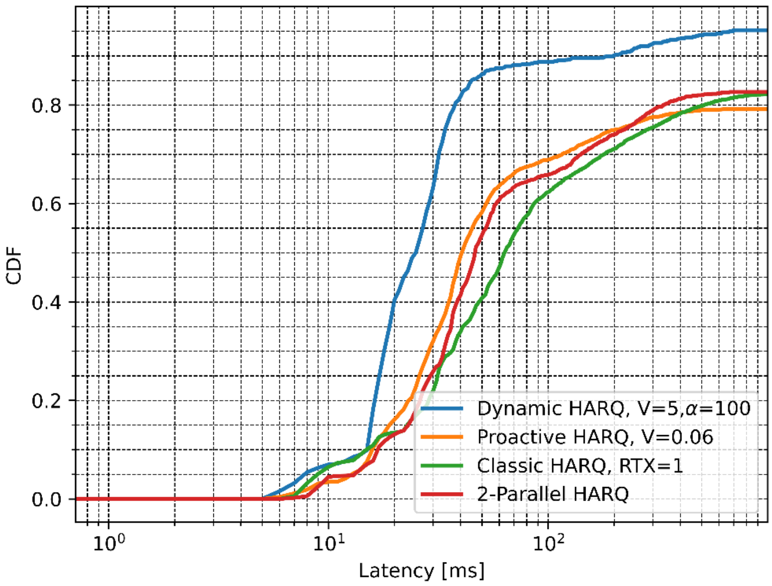


Figure 3.5-9 – Cumulative Distribution Function of RLC Latency for different HARQ schemes

3.5.5. Conclusion

In this section, we proposed a reliable, resource and delay-optimized scheduling suitable for dynamic scenarios (e.g., random bursty traffic, time-varying channel) based on Lyapunov optimization for Ultra-Reliable Low-Latency Communications. It takes into account the traffic arrival at the network layer, the queue behaviors at the data link layer and the risk that the applied decision might trigger packet loss. Resources are allocated on the fly depending on the status of transmissions and retransmissions queues; on the behaviour of the channel and on the KPI we want to optimize. If we want to optimize the overall throughput we want the scheduler to allocate the least resources possible. If we want to maximize the reliability, we want the scheduler to allocate as many resources as possible. The trade-off between the resource efficiency, latency and reliability is achieved by the timing and intensity of decisions and can be parameterized with V and α. Our queue-aware and channel-aware solution has been implemented in an experimental testbed using OpenAirInterface and performance is evaluated with an extensive experimentation in an environment with one gNB and one user. In future work, we plan to increase the number of user, and to apply to them different traffic and more realistic channel thanks to a channel emulator.

3.6. Error Structure Aware Parallel BP-RNN Decoders for Short LDPC Codes

3.6.1. Motivation and related work

Short-packet machine-to-machine communications, central to the emerging Internet of Things (IoT) technology, have revitalized interest in research and practice of efficient error correcting codes, for messages ranging from a few tens up to a few hundred bits. While important progress has been made over the last years in understanding the limits of coding at short block lengths [PPV10], the design of efficient short codes and decoding algorithms still raises many challenges [CDJ+19].

Low Density Parity Check (LDPC) codes [Gal63] are a class of error correcting codes defined by sparse bipartite graphs [Tan81]. They are well-known for their excellent error correction performance at long block lengths, achieving near Shannon channel capacity performance under Belief Propagation (BP) decoding, in the asymptotic limit of the code length [RSU01]. For codes defined by cycle-free bipartite graphs, BP decoding outputs the maximum a posteriori estimates of the coded bits [Wib96]. Although bipartite graphs associated with practical codes contain cycles, BP decoding may still take effective advantage of sparse, long enough graphs, devoid of short cycles. However, for short codes, short cycles may not be avoidable, thus significantly degrading the BP performance. This is even more pronounced for High Density Parity Check (HDPC) codes, defined by higher density bipartite graphs [DB09].

To reduce the impact of short cycles, a weighted BP decoding has been introduced in [NML18], where the weights are optimized using a Neural Network (NN). The topology of the NN mimics the BP decoding process, with unwrapped decoding iterations. The approach may use either a feedforward (FF) or a Recurrent NN (RNN). The corresponding decoders are termed as BP-FF and BP-RNN. It has been shown in [NML18] that the BP-RNN is able to outperform the usual BP decoder for short Bose-Chaudhuri-Hocquenghem (BCH) codes, belonging to the class of HDPC codes. Subsequently, several variants of NN-based BP decoding have been proposed in the literature. [XVTL20] proposed the design of new decoding rules for finite alphabet iterative decoders, based on a quantized NN model. [BHP+20] developed a pruning method of irrelevant check nodes in a neural BP model, aimed at jointly optimizing the code construction and the decoding. In [DHM+19], a neural BP decoding approach was proposed for cyclic redundancy check (CRC)-assisted polar codes.

In this work, we focus on BP-RNN decoding of short block length LDPC codes. To improve the decoding performance, our approach aims at specializing BP-RNN decoders to difficult error events. To do so, we first propose a classification of the error events, according to the structure of the induced subgraph. The classification is driven by the impact of the induced sub-graph on the BP decoding performance. Then, a parallel construction, comprising several BP-RNN decoders running in parallel, is described, where each BP-RNN is specialized to a specific error class. Finally, we discuss the training of the parallel BP-RNN decoders, and provide a method to reduce their number, without jeopardizing the decoding performance, by introducing a similarity rate metric.

3.6.2. System model

We consider an LDPC code defined by a Tanner (bipartite) graph with N variable-nodes and M check-nodes, denoted respectively by $n \in \{1, \dots, N\}$ and $m \in \{1, \dots, M\}$. We further denote by $\mathcal{N}(m)$ the set of variable-nodes connected to a check-node m, and by $\mathcal{M}(n)$ the set of check-nodes connected to a variable-node n.

BP decoding consists of an iterative exchange of messages along the edges of the Tanner graph, where each message provides an estimation of the incident variable-node. BP-RNN and BP-FF decoding algorithms are weighted variants of the BP decoding, where exchanged messages are multiplied by weights learned through an either RNN or FF-NN approach. The underlying NN contains three types of *neural layers*, each one corresponding to a step of the BP algorithm. The *check-pass layer* and the *data-pass layer* carry out the computation of messages outgoing from check-nodes and variable-nodes, respectively. Each one of them contains a number of neurons equal to the number of edges of the Tanner graph. In addition, the *a posteriori Log Likelihood Ratio (LLR) layer* consists of N neurons, computing the a posteriori LLR values of the N variable-nodes. The three layers of the NN are connected such that a check pass layer, followed by a data pass layer and an a posteriori LLR layer model one iteration of the BP decoding. In particular, it is worth stressing out the differences between the edges of the Tanner graph (corresponding to neurons in the check-pass and data-pass layers), and the edges of the NN.

The formulas below detail the calculation of messages within each layer. We denote by $\beta_{m\to n}$ and $\alpha_{n\to m}$ the messages computed by the check-pass and data-pass layers (where (m, n) is an edge of the Tanner graph), and by $\tilde{L}_n$ the messages computed by the a posteriori LLR layer. The observed (channel) LLR values are denoted by $L_{\mathrm{ch},n}$, and are used to initialize $\alpha_{n\to m}$ messages prior to the first iteration.

$$\beta_{m\to n} = 2\tanh^{-1}\left(\prod_{n'\in\mathcal{N}(m)\backslash n} \tanh\left(\frac{\alpha_{n'\to m}}{2}\right)\right) \tag{3.6-1}$$

$$\alpha_{n\to m} = L_{\mathrm{ch},n} + \sum_{m'\in\mathcal{M}(n)\backslash\{m\}} w_{m'\to n\to m}\beta_{m'\to n} \tag{3.6-2}$$

$$\tilde{L}_n = L_{\mathrm{ch},n} + \sum_{m\in\mathcal{M}(n)} \widetilde{w}_{m\to n}\beta_{m\to n} \tag{3.6-3}$$

It can be observed that weights are applied only on the NN edges incoming to the data-pass (3.6-2) and a posteriori LLR (3.6-3) layers. Each weight corresponds to one specific edge of the NN. In (3.6-2) the weights are denoted by $w_{m'\to n\to m}$, where the subscript indicates both the corresponding neuron $n \to m$ in the data-pass layer, and the incoming NN edge from neuron $m' \to n$ in the check-pass layer. In (3.6-3) the weights are denoted by $\widetilde{w}_{m\to n}$, where the subscript indicates the corresponding neuron n in the a posteriori LLR layer, and the incoming NN edge from neuron $m \to n$ in the check-pass layer. For the BP-RNN, the weights only depend on the corresponding edges of the NN, while for the BP-FF they also depend on the iteration number (note that, to simplify notation, we have not indicated the iteration number on the above formulas). It is worth noticing that despite the reduced number of trained weights, the BP-RNN achieves similar performance to the BPFF [8].

An alternative approach suggested in [8] to further reduce the number of weights is based on the following data-pass layer:

$$\alpha_{n\to m} = L_{\mathrm{ch},n} + w_{n\to m}\sum_{m'\in\mathcal{M}(n)\backslash\{m\}} \beta_{m'\to n} \tag{3.6-4}$$

where the applied weight only depends on the data-pass neuron. This simplification reduces the training complexity and makes it possible to reuse conventional BP decoding architectures for efficient hardware implementation. The BP-RNN using (3.6-4) will be referred to as BP-RNN Hardware Friendly Implementation (BP-RNN-HFI). To train the BP-RNN, we use the following Bit Error Rate (BER) loss function, assuming without loss of generality that the zero codeword is transmitted:

$$\mathrm{Loss}(\hat{L}) = \frac{-1}{N}\sum_{n=1}^{N} \log\left(\sigma(\tilde{L}_n)\right) \tag{3.6-5}$$

where $\sigma(x) = \left(1 + \exp(x)\right)^{-1}$ is the sigmoid function, converting the LLRs into probability values. The loss function is minimized during the NN training, thus improving the BER of the trained decoder.

3.6.3. Specializing BP-RNN decoders according to an error event classification

3.6.3.1. Absorbing type classification of error events

Let V be a set of variable-nodes, and C be the set of check-nodes connected to at least one variable-node in V. We denote by $O(V) \subset C$ the set of check-nodes connected an odd number of times to V (that is, they have odd degree in the sub-graph induced by V). Thus, $E(V) \coloneqq C\backslash O(V)$ is the set of check-nodes connected an even number of times to V. The set V is said to be an absorbing set [DLZ+09], if each variable-node in V has fewer neighbours in $O(V)$ than in $E(V)$. Figure 3.6-1 shows an example of absorbing set (left case), where each variable-node of V is connected to one check-node in $O(V)$ and two check-nodes in $E(V)$. While absorbing sets are combinatorial substructures of the Tanner graph, defined independently of the particular decoding algorithm, they are known to be particularly harmful to BP or other forms of message-passing decoding. Indeed, assuming that the set of errors V is an absorbing set, then each variable-node in V has less neighbour check-nodes indicating an error (unsatisfied), than indicating no error (satisfied). Consequently, V represents a difficult error event, yielding a decoding failure with high probability.

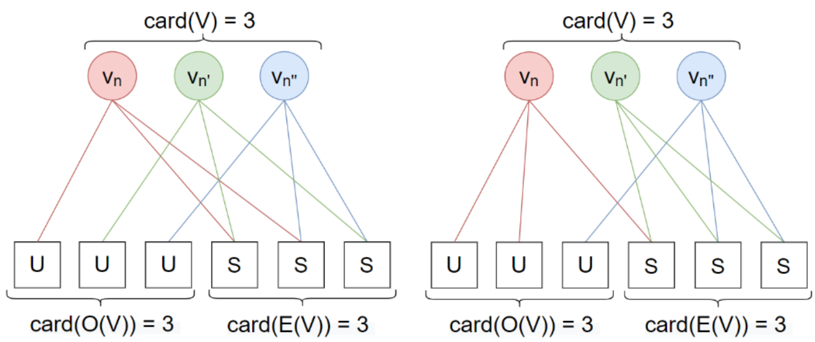


Figure 3.6-1 – Example of two sets V, with $card(O(V)) = card(E(V)) = 3$. If variable-nodes in V are in error, check-nodes marked by an U are unsatisfied, while those marked by an S are satisfied. The left case corresponds to the absorbing set case, while the right one is a non-absorbing set case

We are interested in classifying error events with a given number of errors ν. Let V a set of variable-nodes, with $card(V) = \nu$. We define the absorbing type of V as the pair (ω, ε), where $\omega := \mathrm{card(O(V))}$ and $\varepsilon := \mathrm{card(E(V))}$. We shall sometimes denote the absorbing type as $\nu - (\omega, \varepsilon)$, to also account for the cardinality of V. Note that the absorbing type indicates the total number of (unsatisfied, satisfied) check-nodes, in case the variable-nodes in V are in error. However, variable-node sets of same absorbing type may induce different (precisely, non-isomorphic) sub-graphs. Such an example is illustrated in Figure 3.6-1 for two variable-node sets of absorbing type $3 - (3, 3)$, the first of which is an absorbing set (left), but not the second (right). For the set V in the right case, it can be seen that variable-node n is connected to two unsatisfied check-nodes and one satisfied (most favourable case among the three variable-nodes), n' to only satisfied check-nodes (worst case), and n'' to one unsatisfied and two satisfied check-nodes. If variable-node n gets corrected, V reduces to an absorbing set of type $2 - (2, 2)$, determined by n' and n''. However, in general there is no guarantee that variable-node n can be decoded by the BP decoder (this will depend on the noise model, and the actual noise realization), thus we consider the right case as an intermediate case, lying between the $3 - (3, 3)$ absorbing set case and the $2 - (2, 2)$ absorbing set case.

Accordingly, we define the error class $\nu - (\omega, \varepsilon)$ as comprising all the error events, whose underlying variable-node error set V has absorbing type $\nu - (\omega, \varepsilon)$. We further partition the above error class into error sub-classes, with each sub-class corresponding to variable-node error sets V of absorbing type $\nu - (\omega, \varepsilon)$, and inducing isomorphic sub-graphs. Sub-classes are denoted by $\nu - (\omega, \varepsilon, s)$, where s denotes the sub-class index. Accordingly, the variable-node sets illustrated in Figure 3.6-1 are associated with the sub-classes $3 - (3, 3, 1)$ and $3 - (3, 3, 2)$. In the sequel, we shall simply refer to $\nu - (\omega, \varepsilon, s)$ as error classes (rather than sub-classes), since no confusion is possible.

3.6.3.2. Proposed parallel BP-RNN decoders

To improve the decoding performance of LDPC codes at short coding length, we propose to specialize (i.e., train) a BP-RNN decoder for each error class $\nu - (\omega, \varepsilon, s)$, according to the classification from the previous subsection. The number of different error classes, and thus of BP-RNN decoders, depend on the particular Tanner graph defining the LDPC code, and the value of ν. For a given short LDPC codes, we determine all the possible error classes, by considering all the variable-node subsets V of cardinality ν. In this work, we consider $\nu = 2{,}3$, thus the proposed approach is particularly relevant to high coding rate LDPC codes, correcting a small number of errors. Once a BN-RNN decoder has been trained for each error class (the training procedure will be detailed in next subsection), we consider a parallel decoding architecture, where all the trained decoders are run in parallel (note that different architectural choices are possible and not discussed in this paper). We include the conventional BP decoder in the parallel structure. If none of the parallel decoders outputs a codeword (which is verified by computing the syndrome), decoding fails. Otherwise, among the decoded codewords, we select the one that has been outputted the most often (in case different decoders output different codewords). Decoding is successful if the selected codeword is equal to the transmitted one.

3.6.3.3. Training of the parallel BP-RNN decoders

We propose in this section a construction of the training set, used to train the BP-RNN decoder for a particular error class. We assume that coded bits are mapped to ± 1 modulated symbols, which undergo real additive white Gaussian noise (AWGN). Since both the noise model and the BP-RNN decoder are symmetric [NML18], we may assume the all-zero codeword is transmitted, corresponding to an all $+1$ modulated signal. Hence, under the AWGN model, received symbols are given by $y_n = 1 + z_n$, $n = 1, \dots, N$, where z_n denotes a real-valued normal distributed random variable, with mean 0, and variance σ^2. To generate a random error event in a given error class $\nu - (\omega, \varepsilon, s)$, we first consider an underlying variable-node error set V, randomly chosen from those corresponding to the given error class, and then generate received symbols y_n, by

$$y_n = 1 + z_n, \forall\, n = 1, \dots, N \tag{3.6-6}$$

$$\text{where } z_n \hookrightarrow \begin{cases} \mathcal{N}(0, \sigma^2, -\infty, -1), \text{ if } n \in V \\ \mathcal{N}(0, \sigma^2, -1, \infty), \text{ otherwise} \end{cases} \tag{3.6-7}$$

where $\mathcal{N}(0, \sigma^2, a, b)$ denotes the truncated normal distribution with mean 0 and variance σ^2, taking values in the interval (a, b). The training set is obtained by repeating the above procedure multiple times, for each variable-node set V in the given error class. In this way, the training set is representative of the error class, and thus the trained BP-RNN decoder becomes specialized to error events in the class.

3.6.3.4. Complementary selection of trained BP-RNNs

In practical applications, it is desirable to reduce the number of decoders running in parallel. To this end, in this subsection we propose a complementarity metric between the trained decoders. Intuitively, two decoders are complementary if the probability to fail on the same error event is low. This maximizes the gain when the two decoders are run in parallel. We define the similarity rate between two decoders D_p and D_q, $p \neq q$, as the probability that both decoders fail, when either one of them fails. Precisely, we define

$$S(D_p, D_q) \coloneqq \Pr\left(D_p \text{ and } D_q \text{ fail} \mid \text{either } D_p \text{ or } D_q \text{ fails}\right) \tag{3.6-8}$$

In practice, this metric can be numerically estimated by Monte-Carlo simulation, using

$$S(D_p, D_q) \approx \frac{N_{p,q}}{N_p + N_q - N_{p,q}} \tag{3.6-9}$$

where N_p (resp. N_q) is the number of times the decoder D_p (resp. D_q) failed, and $N_{p,q}$ is the number of times they both failed. Consequently, $S(D_p, D_q)$ measures the effectiveness of specialized training in producing decoders able to correct different error events. Put differently, it characterizes the level of complementary between the two decoders. An overall similarity coefficient is then calculated for each decoder D_p, by averaging the similarity rate between D_p and the other decoders.

$$\bar{S}(D_p) = \frac{1}{D-1} \sum_{p \neq q} S(D_p, D_q) \tag{3.6-10}$$

where D denotes the total number of parallel BP-RNN decoders. Subsequently, given a similarity threshold value S_{th}, we keep only the decoders with overall similarity coefficient $\bar{S}(D_p)$ less than S_{th}. Hence, only the most complementary decoders are maintained in the final parallel structure. We note that the choice of the S_{th} value may yield different trade-offs between complexity and decoding performance.

3.6.4. Numerical results

3.6.4.1. Simulation Settings

Two LDPC codes with regular variable-node degree have been considered in our simulations. Code parameters are provided in Table 2.7-1, where N denotes the code length, K the number of information bits, $R_c \coloneqq K/N$ the coding rate, d_v the variable nodes degree, d_c the check nodes degree, and $n_{4-\text{cycles}}$ the number of length-4 cycles.

Table 2.7-1 – Parameters of the constructed codes

	N	K	R_c	d_v	d_c	$n_{4-\text{cycles}}$
Code-1	64	46	0.71	3	10-11	47
Code-2	128	105	0.81	3	16-17	1130

The Tanner graphs of the two codes have been constructed by using the Progressive Edge Growth (PEG) algorithm [HEA05], a greedy algorithm making the best effort to reduce the number of short cycles in the constructed graph. Yet, for small code length, short cycles, including cycles of length-4, cannot be completely avoided. The number of decoding iterations was set to ten, for all the decoders. Since we consider codes with high coding rate, the error events classification was conducted for error sets V of size $\nu = 2, 3$[5]. Applying the error classification procedure described in Section 2.7.3.1, we found three $2-(\omega,\varepsilon,s)$ classes, for both Code-1 and Code-2, and ten (resp. eleven) $3-(\omega,\varepsilon,s)$ error classes for Code-1 (resp. Code-2). Thus, a total of thirteen (resp. fourteen) BP-RNNs were used in parallel, alongside the BP decoder. Each BP-RNN was trained independently with the training set construction technique described in Section 2.7.3.3. The same procedure was repeated for the BP-RNN-HFI decoder, for both Code-1 and Code-2, in order to assess the weight reduction in (2.7-4). In addition, we also trained a single BP-RNN decoder according to the procedure described in [NML18], to provide benchmark for our parallel BP-RNNs approach. To train the BP-RNN decoders, we used the Keras library, with the hyper parameters shown in Table 2.7-2.

Table 2.7-2 – Keras parameters

Parameters	Parameters values
Optimizer (Gradient descent)	RMSprop [TH12] (initialized at a learning rate of 10^{-3})
Epoch number	10
Training batch size	8192
Test batch size	16384

All BP-RNNs were trained for each signal-to-noise ratio (SNR) value ranging from 1 dB to 8 dB, with a step of 1 dB, thus providing eight optimized weight sets for each decoder. While this increases the training complexity, it also improves the decoding performance, as compared to the case where only one training is performed, mixing together with all SNR values. During the simulations (i.e., test stage, after the training was performed), each weight set was used for the corresponding SNR value, except near 9 and 8.5 dB where the weights of 8 dB were used.

Finally, we used the similarity metric introduced in Section 2.7.3.4, in order to reduce the number of parallel decoders. The similarity coefficient has been numerically estimated by using (2.7-9) and (2.7-10), based on the first simulation results. We fixed a similarity threshold value $S_{\text{th}} = 0.6$ (resp. $S_{\text{th}} = 0.56$) for Code-1 (resp. Code-2), at a Frame Error Rate (FER) of 10^{-4}, which led to the selection of only eight (resp. nine) trained BP-RNN decoder alongside the conventional BP. Then, new independent simulations were run for both resulting parallel structures, to assess their FER performance.

[5] For comparison, a BCH code with either $(N,K) = (63,45)$ or $(N,K) = (127,106)$ has minimum distance $d = 7$, thus may correct 3 errors

3.6.4.2. FER performance

We compare the different decoding strategies discussed in the previous section, in terms of FER. Reported SNR gains are evaluated at a FER of 10^{-4}.

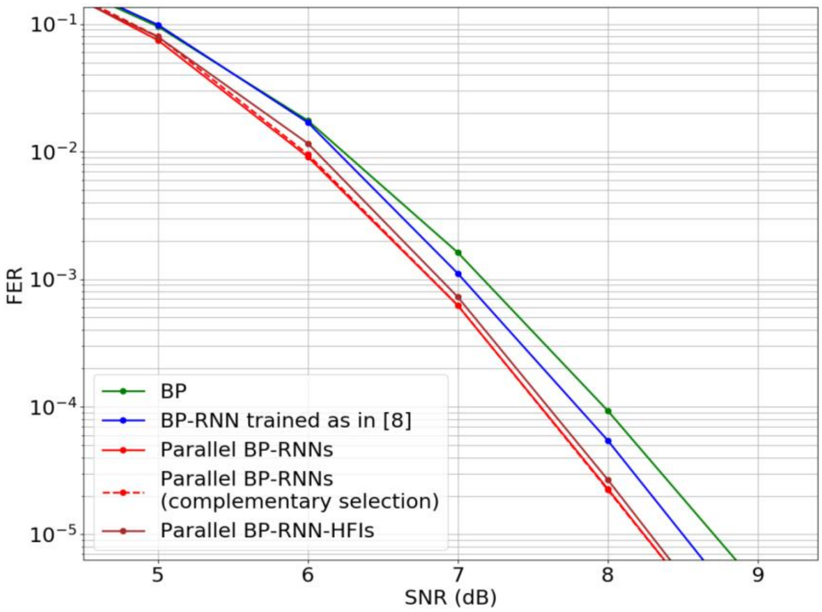


Figure 3.6-2 – FER results for Code-1 (in the legend, [8] is [NML18])

Simulation results for Code-1 are shown in Figure 3.6-2. The BP-RNN trained as in [NML18] yields an SNR gain of 0.18 dB, with respect to conventional BP. Using the proposed structure, with specialized BP-RNNs, the SNR gain is increased to 0.42 dB. The parallel BP-RNN-HFIs exhibit only negligible performance degradation compared to the parallel BP-RNNs, due to the weight reduction constraint in (2.7-4). Finally, it can be observed that the parallel BP-RNNs construction with complementary selection shows virtually the same FER performance as the original complete construction of parallel BP-RNNs. Therefore, choosing $S_{\mathrm{th}} = 0.6$ leads to a selection of eight decoders, which effectively complement each others.

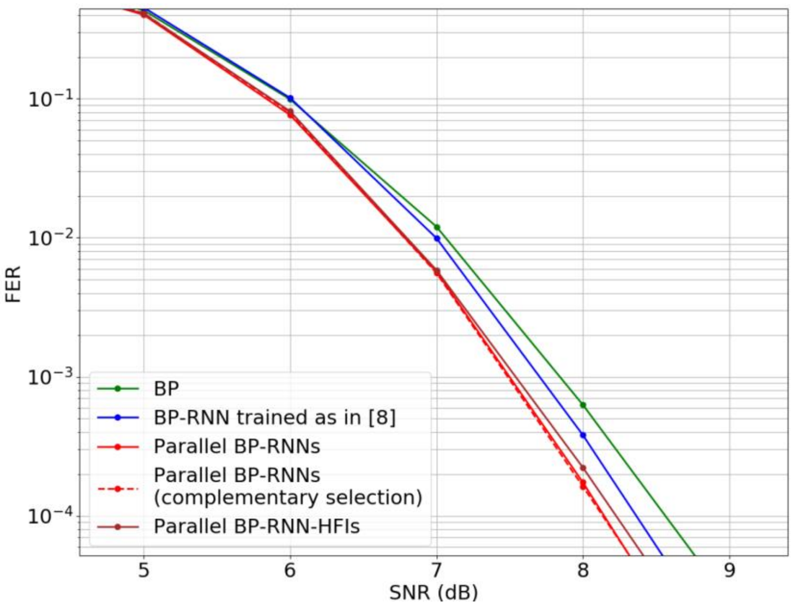


Figure 3.6-3 – FER results for Code-2 (in the legend, [8] is [NML18])

Simulation results for Code-2 are shown in Figure 3.6-3. Despite the increase in the length of the code, it should be noted that Code-2 exhibits an increased number of length-4 cycles, due to its higher coding rate. However the parallel BP-RNN decoder yields a similar SNR gain, of 0.42 dB, with respect to the conventional BP. The parallel BP-RNN-HFIs tend to be a little less efficient. Furthermore, Fig. 2.7-3 also corroborates the relevance of the complementary selection, the corresponding parallel BP-RNNs construction yielding again almost the same performance as the original complete construction of parallel BP-RNNs.

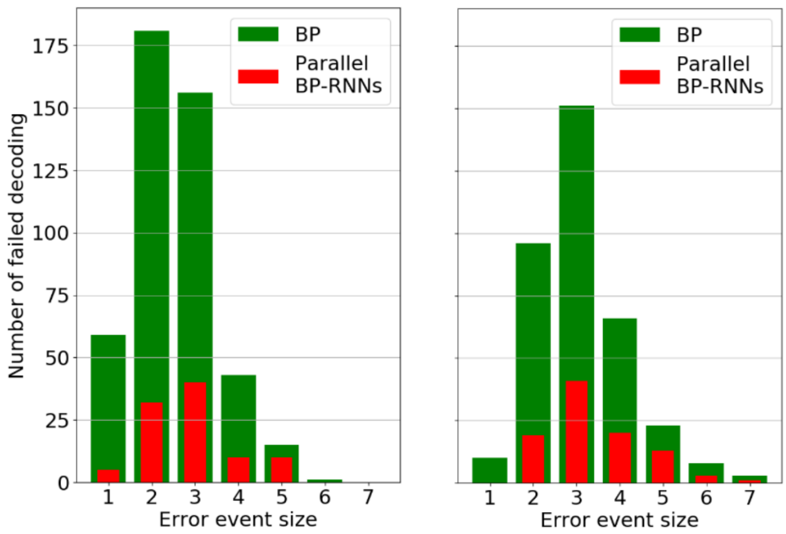


Figure 3.6-4 – Number of failed decoding vs. the error event size, for Code-1 (left) and Code-2 (right)

Finally, Figure 3.6-4 provides a comparison in terms of the number of decoding failures, for various values of ν (size of the error event), for an SNR of 8 dB. It demonstrates that for both Code-1 and Code-2, the parallel BP-RNNs are clearly able to correct numerous error events which are not decoded by the BP, especially for the two and three error events ($\nu = 2, 3$). Therefore, we conclude that the specialization of the training for the $2 - (\omega, \varepsilon, s)$ and the $3 - (\omega, \varepsilon, s)$ error classes effectively induces an understanding of the BP-RNNs of how to decode several type of size two and three error events. Furthermore, some error events of size four and five are also successfully decoded thanks to the previous specialization.

3.6.5. Conclusion

In this paper, we addressed the problem of enhancing the BP-RNN performance at short coding length. To this end, we studied and classified error events according to the impact of the induced sub-graphs on the BP decoding performance. Then, we proposed a new decoding strategy consisting of parallel specialized BP-RNN decoders where each BP-RNN was trained for a specific error class. In addition, we introduced a complementary selection method of the trained BP-RNN decoders, proven to be efficient in keeping the most relevant trained decoders in the final parallel structure.

This work is a first step towards a framework of specialized neural BP decoders, and we believe that further work may reveal alternative specialization strategies. The final aim would be to approach maximum likelihood decoding performance at short to moderate code-length, for which we will probably need to rely on a bunch of practical decoders, rather than a unique one.

4. CONCLUSION

CPSs will play a more and more prominent role in many strategic industrial sectors and European SMEs as well as big groups will benefit from the exploitation of CPS technologies from various points of view: CPSs will contribute to improve companies' productivity, diminish their operational costs, and support the optimization of their logistics. By their very own nature of highly interconnected systems both among themselves and with the surrounding environment, CPSs need novel dedicated communication solutions, suitable for a completely new variety of applications and requirements. These solutions may partially come from a straightforward adaptation or application of already standardized technologies, but in non-negligible part require an innovation effort that goes beyond what 4G and 5G mobile networks currently offer.

In line with the vision of "beyond-5G" networks that standardization bodies are starting to develop, in this report we described the research studies carried out so far by Task 2.2 of CPS4EU. This work is grounded on the motivation and use case requirements highlighted by Task 2.1 and concerns both CPSs' communication modules and their entire supporting network infrastructure. Our opinion is that an efficient integration of CPS technologies into the European industrial scenario must rely on fast and reliable communications. For this reason, our contributions specifically focused on URLLC and TSN.

Our main conclusion is that the conception of CPS-adapted communication networks and technologies will be effective if it happens in the most holistic and inter-layer manner possible: it will have to combine new networking solutions for xHauling with physical-layer innovation and optimized resource allocation and management schemes for all the involved network elements. The system or network models, the algorithms, and the numerical results described and summarized in this report strengthen this vision and provide promising tools for the future development of CPS technologies. Part of the solutions developed in Task 2.2 have been also transferred to Task 2.3, and further implementation and evaluation results have been provided in deliverables [D2.5] and [D2.6].

It must be noted as well that this work has enabled a fruitful collaboration between the partners inside WP2 as well as with other WPs. This has triggered up to nine (9) publications and three (3) patents. Several tools like the VSORA DSP + Toolchain and the sub prototypes can be exploited and/or re used immediately and after the end of the project.

5. REFERENCES

[AA19a] A. Alizadeh and M. Vu, “Load balancing user association in millimeter wave MIMO networks,” *IEEE Trans. on Wireless Commun.*, vol. 18, no. 6, pp. 2932-2945, June 2019.

[AA19b] A. Alizadeh and M. Vu, “Early acceptance matching game for user association in 5G cellular HetNets,” Proc. GLOBECOM, Waikoloa, Hawaii, USA, pp. 1-6, Dec. 2019.

[ADD17] A. De Domenico *et al.*, “Making 5G millimeter-wave communications a reality [Industry Perspectives],” *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 4-9, Aug. 2017.

[ANKB16] K. Alexandris, N. Nikaein, R. Knopp, and C. Bonnet, “Analyzing x2 handover in LTE/LTE-A,” in *14th IEEE International Symposium on Modeling and Optimization in Mobile, Ad-hoc, and Wireless Networks (WiOpt)*, pp. 1– 7, 2016.

[AV17] A. Vaswani *et al.*, “Attention is all you need,” *Proc. NIPS*, Long Beach, California, USA, 2017, pp. 5998–6008.

[BHP+20] A. Buchberger, C. Hager, H. D. Pfister, L. Schmalen, and A. G. i Amat, “Pruning neural belief propagation decoders,” in *IEEE International Symposium on Information Theory (ISIT)*, 2020, pp. 338–342.

[BJO20] Björnson, E. & Sanguinetti, L., “Scalable Cell-Free Massive MIMO Systems”, in *IEEE Transactions on Communications*, 2020, 68, 4247-4261.

[BJO20a] Björnson, E. & Sanguinetti, L., “Making Cell-Free Massive MIMO Competitive With MMSE Processing and Centralized Implementation”, in *IEEE Transactions on Wireless Communications*, 2020, 19, 77-90.

[BDF20] S. Bicaïs, J.-B. Doré, A. Falempin, “Méthode de démodulation par apprentissage automatique pour récepteurs MIMO à détection d’énergie, ” patent proposal n. FR2005670, submitted on May 29th, 2020.

[BFD+22] S. Bicaïs, A. Falempin, J.-B. Doré, V. Savin, “Design and analysis of MIMO systems using energy detectors for sub-THz applications,” *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, 2022.

[CDJ+19] M. C. Coskun, G. Durisi, T. Jerkovits, G. Liva, W. Ryan, B. Stein, and F. Steiner, “Efficient error-correcting codes in the short blocklength regime,” *Physical Communication*, vol. 34, pp. 66–79, 2019.

[D2.1] CPS4EU, Deliverable 2.1, “Specification and architecture of the communication modules”, Dec. 2019.

[D2.2] CPS4EU, Deliverable 2.1, “Specification and architecture of the communication modules – v2”, Nov. 2020.

[D2.3] CPS4EU, Deliverable 2.3, “Propositions for 5G, including URLLC evolution - v1”, Dec. 2020.

[D2.4] CPS4EU, Deliverable 2.4, “Propositions for 5G, including URLLC evolution – v2”, Aug. 2022.

[D2.5] CPS4EU, Deliverable 2.5, “Simulation tools and experimental platforms – v1”, Dec. 2020.

[D2.6] CPS4EU, Deliverable 2.6, “Simulation tools and experimental platforms – v2”, Sept. 2022.

[DAW06] Dawod, N. H.; Marsland, I. D. & Hafez, R. H. M., “Improved transmit null steering for MIMO-OFDM downlinks with distributed base station antenna arrays”, in *IEEE Journal on Selected Areas in Communications*, 2006, 24, 419-426.

[DB09] I. Dimnik and Y. Be’ery, “Improved random redundant iterative HDPC decoding,” *IEEE Transactions on Communications*, vol. 57, no. 7, pp. 1982–1985, 2009.

[DHM+19] N. Doan, S. A. Hashemi, E. N. Mambou, T. Tonnellier, and W. J. Gross, “Neural belief propagation decoding of CRC-polar concatenated codes,” in *IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.

[DL16] D. Liu *et al.*, “User association in 5G networks: A survey and an outlook,” *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1018-1044, 2016.

[DLMC22] L. N. Dinh, I. Labriji, M. Maman, and E. Calvanese Strinati, "Toward URLLC with Proactive HARQ Adaptation," *European Conference on Networks and Communications (EUCNC)*, 2022.

[DLZ+09] L. Dolecek, P. Lee, Z. Zhang, V. Anantharam, B. Nikolic, and M. Wainwright, "Predicting error floors of structured LDPC codes: Deterministic bounds and estimates," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 6, pp. 908–917, 2009.

[ECS19] E. Calvanese Strinati *et al.*, "6G: The next frontier: From holographic messaging to artificial intelligence using subterahertz and visible light communication," *IEEE Veh. Technol. Mag.*, vol. 14, pp. 42-50, Sep. 2019.

[EPEL20] E. Peltonen *et al.*, "6G white paper on edge intelligence," *arXiv preprint*, Apr. 2020. [Online]. Available: https://arxiv.org/abs/2004.14850.

[EW20] E. Wijmans *et al.*, "DD-PPO: Learning near-perfect point goal navigators from 2.5 billion frames," *Proc. ICLR*, Addis Ababa, Ethiopia, Apr.-May 2020.

[GA15] G. Athanasiou, P. C. Weeraddana, C. Fischione, and L. Tassiulas, "Optimizing client association for load balancing and fairness in millimeter-wave wireless networks," *IEEE/ACM Trans. Netw.*, vol. 23, pp. 836-850, June 2015.

[Gal63] R. G. Gallager, "Low density parity check codes," MIT Press, Cambridge, 1963, research Monograph series.

[HEA05] X.-Y. Hu, E. Eleftheriou, and D.-M. Arnold, "Regular and irregular progressive edge-growth Tanner graphs," *IEEE Transactions on Information Theory*, vol. 52, no. 51, pp. 386–398, 2005.

[HSC+15] D. Han, S. Shin, H. Cho, J.-M. Chung, D. Ok, and I. Hwang, "Measurement and stochastic modeling of handover delay and interruption time of smartphone real-time applications on LTE networks," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 173–181, 2015.

[JAB+17] T. Jacobsen, R. Abreu, G. Berardinelli, K. Pedersen, P. Mogensen, I. Z. Kovacs, and T. K. Madsen, "System Level Analysis of Uplink Grant-Free Transmission for URLLC," in *IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, 2017.

[JBSL16] M. Jabi, M. Benjillali, L. Szczecinski, and F. Labeau, "Energy Efficiency of Adaptive HARQ*," IEEE Transactions on Communications*, vol. 64, no. 2, pp. 818–831, 2016.

[JDC20] J.-L. Jimenez Gonzalez, C. Dehos, N. Cassiau, "Channel bonding transceivers for 6G future networks," in *Proc. 6G Summit*, 2020.

[JDPM16] L. Jing, E. De Carvalho, P. Popovski, A. O. Martinez, "Design and performance analysis of noncoherent detection systems with massive receiver arrays," *IEEE Trans. Signal Process.*, vol. 64, no. 19, pp. 5000-5010, 2016.

[JS17] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint*, July 2017. [Online]. Available: https://arxiv.org/abs/1707.06347.

[JS18] J. Schulman, P. Moritz, S. Levine, M. I. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation", *arXiv preprint*, Oct. 2018. [Online]. Available: https://arxiv.org/abs/1506.02438.

[KHO19] Khoshnevisan, M.; Joseph, V.; Gupta, P.; Meshkati, F.; Prakash, R. & Tinnakornsrisuphap, P., "5G Industrial Networks With CoMP for URLLC and Time Sensitive Network Architecture", in *IEEE Journal on Selected Areas in Communications*, 2019, 37, 947-959.

[KVIT04] B. Kang, N. Vijaykrishnan, M. J. Irwin, and T. Theocharides, "Powerefficient implementation of turbo decoder in sdr system," in *Proc. Of IEEE International SOC Conference*, pp. 119–122, 2004.

[LB08] L. Buşoniu, R. Babuŝka, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Trans. Syst., Man, Cybern. C Appl. Rev.*, vol. 38, no. 2, pp. 156-172, Mar. 2008.

[LDE+21] Y. Liu, Y. Deng, M. Elkashlan, A. Nallanathan, and G. K. Karagiannidis, "Analyzing Grant-Free Access for URLLC Service," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 3, pp. 741–755, 2021.

[LI01] Jian Li; Guoqing Liu & Giannakis, G. B., “Carrier frequency offset estimation for OFDM-based WLANs”, *IEEE Signal Processing Letters*, 2001, 8, 80-82.

[LM07] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, “Hysteretic Q-learning: An algorithm for decentralized reinforcement learning in cooperative multi-agent teams,” *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, California, USA, Dec. 2007, pp. 64-69.

[LSK21] T.-K. Le, U. Salim, and F. Kaltenberger, “An Overview of Physical Layer Design for Ultra-Reliable Low-Latency Communications in 3GPP Releases 15, 16, and 17,” *IEEE Access*, vol. 9, pp. 433–444, 2021.

[MAA16] H. Mukhtar, A. Al-Dweik, and M. Al-Mualla, “Content-aware and occupancy-based hybrid ARQ for video transmission,” in *IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 1–4, 2016.

[MBHS20] J. Manco, G. G. Banos, J. Haarri, and M. Sepulcre, “Prototyping v2x applications in large-scale scenarios using openairinterface,” *IEEE Vehicular Networking Conference (VNC)*, pp. 1–4, 2020.

[MS20] M. Sana, A. De Domenico, W. Yu, Y. Lostanlen, and E. Calvanese Strinati, “Multi-Agent Reinforcement Learning for Adaptive User Association in Dynamic mmWave Networks,” IEEE Transactions on Wireless Communications, vol. 19, no. 10, pp. 6520–6534, 2020.

[NEE10] N.J. Neely, “Stochastic Network Optimization with Application to Communication and Queueing Systems”, *Synthesis Lectures on Communication Networks*, vol. 3, 2010

[NGO17] Ngo, H. Q.; Ashikhmin, A.; Yang, H.; Larsson, E. G. & Marzetta, T. L., “Cell-Free Massive MIMO Versus Small Cells”, in *IEEE Transactions on Wireless Communications*, 2017, 16, 1834-1850.

[NML18] E. Nachmani, E. Marciano, L. Lugosch, W. J. Gross, D. Burshtein, and Y. Be’ery, “Deep learning methods for improved decoding of linear codes,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 119–131, 2018.

[NMM+14] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, “Openairinterface: A flexible platform for 5g research,” *ACM SIGCOMM Computer Communication Review*, pp. 33–38, 2014.

[NZ18] N. Zhao, Y. Liang, D. Niyato, Y. Pei, and Y. Jiang, “Deep reinforcement learning for user association and resource allocation in heterogeneous networks,” *Proc. IEEE GLOBECOM*, Abu Dhabi, UAE, Dec 2018, pp. 1-6.

[PAN06] Pantos, G. D., “A Numerical Technique for Blind Estimation of Carrier Frequency Offset in OFDM Systems”, in *IEEE Transactions on Broadcasting*, 2006, 52, 566-569.

[PDN14] S. Pfletschinger, D. Declercq, and M. Navarro, “Adaptive HARQ With Non-Binary Repetition Coding,” *IEEE Transactions on Wireless Communications*, vol. 13, no. 8, pp. 4193–4204, 2014.

[PKT+12] J. Park, S. Kang, S. V. Thyagarajan, E. Alon, A. M. Niknejad, “A 260 GHz fully integrated CMOS transceiver for wireless chip-to-chip communication,” in *Proc. VLSIC*, Honolulu, HI, USA, pp. 48–49, 2012.

[PPV10] Y. Polyanskiy, H. V. Poor, and S. Verdu, “Channel coding rate in the finite blocklength regime,” *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, 2010.

[PW09] G. K. Psaltopoulos, A. Wittneben, “Diversity and spatial multiplexing of MIMO amplitude detection receivers,” in *Proc. IEEE PIMRC*, Tokyo, Japan, pp. 202–206, 2009.

[PZ19] P. Zhou, X. Fang, X. Wang, Y. Long, R. He, and X. Han, “Deep learning-based beam management and interference coordination in dense mmWave networks,” *IEEE Trans. Veh. Technol.*, vol. 68, no.1, pp. 592-603, Jan. 2019.

[QM18] Q. Mao, F. Hu, and Q. Hao, “Deep learning for intelligent wireless networks: A comprehensive survey,” *IEEE Commun. Surveys Tuts.*, vol. 20, pp. 2595-2621, 2018.

[RAP15] Rappaport, T. S.; MacCartney, G. R.; Samimi, M. K. & Sun, S., “Wideband Millimeter-Wave Propagation Measurements and Channel Models for Future Wireless Communication System Design”, in *IEEE Transactions on Communications*, 2015, 63, 3029-3056.

[RJM17] R. J. Mailloux, *Phased Array Antenna Handbook*. Norwood, MA, USA: Artech House, Inc., 3rd ed., 2017.

[RL20] R. Liu, M. Lee, G. Yu, and G. Y. Li, “User association for millimeterwave networks: A machine learning approach,” *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4162-4174, July 2020.

[RS14] R. Srikant and L. Ying, *Communication Networks: An Optimization, Control and Stochastic Networks Perspective*. USA: Cambridge University Press, 2014.

[RSS18] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. The MIT Press, second ed., 2018.

[RSU01] T. Richardson, M. Shokrollahi, and R. Urbanke, “Design of capacity approaching irregular low-density parity-check codes,” *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 619–637, 2001.

[SJP10] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345–1359, 2010.

[SO17] S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian, “Deep decentralized multi-task multi-agent reinforcement learning under partial observability,” Proc. ICML, Sydney, Australia, Aug. 2017, pp. 2681-2690.

[SOL18] Solomitckii, D.; Orsino, A.; Andreev, S.; Koucheryavy, Y. & Valkama, M., “Characterization of mmWave Channel Properties at 28 and 60 GHz in Factory Automation Deployments”, in Proc. *IEEE Wireless Communications and Networking Conference (WCNC)*, 2018.

[SPE99] Speth, M.; Fechtel, S. A.; Fock, G. & Meyr, H., “Optimum receiver design for wireless broad-band systems using OFDM. I”, in *IEEE Transactions on Communications*, 1999, 47, 1668-1677.

[Tan81] R. Tanner, “A recursive approach to low complexity codes,” *IEEE Transactions on Information Theory*, vol. 27, no. 5, pp. 533–547, 1981.

[TB15] T. Bai and R. W. Heath, “Coverage and rate analysis for millimeter-wave cellular networks,” *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 1100-1114, Feb. 2015.

[TBB20] M. -T. Thi, S. Ben Hadj Said, M. Boc, "SDN-based management solution for time synchronization in TSN networks," in *Proc. IEEE ETFA*, Vienna, Austria, pp. 361-368, 2020.

[TGJ19] M. Tayyab, X. Gelabert, and R. Jantti, “A survey on handover management: From LTE to NR,” *IEEE Access*, vol. 7, pp. 118907–118930, 2019.

[TH12] T. Tieleman and G. Hinton, “Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude,” COURSERA: Neural networks for machine learning, vol. 4, no. 2, pp. 26–31, 2012.

[VPZ13] M. Voicu, D. Pepe, D. Zito, “Performance and trends in millimetre-wave CMOS oscillators for emerging wireless applications,” *International J. Microw. Sci. Technol.*, vol. 2013, pp. 1-6, 2013.

[Wib96] N. Wiberg, “Codes and decoding on general graphs,” Ph.D. dissertation, Department of Electrical Engineering, Linkoping University, Sweden, 1996.

[XVTL20] X. Xiao, B. Vasic, R. Tandon, and S. Lin, “Designing finite alphabet iterative decoders of LDPC codes via recurrent quantized neural networks,” *IEEE Transactions on Communications*, vol. 68, no. 7, pp. 3963–3974, 2020.

[YDG19] Y. Deng, “Deep learning on mobile devices: A review,” *Proc. SPIE 10993*, *Mobile Multimedia/Image Processing, Security, and Applications*, Baltimore, Maryland, USA, May 2019.

[YL18] Y. Liu, X. Fang, M. Xiao, and S. Mumtaz, “Decentralized beam pair selection in multi-beam millimeter-wave networks,” *IEEE Trans. Commun.*, vol. 66, no. 6, pp. 2722-2737, June 2018.

[ZAR18] Zarikoff, B. W. & Cavers, J. K., “Multiple Frequency Offset Estimation for the Downlink of Coordinated MIMO Systems”, in *IEEE Journal on Selected Areas in Communications*, 2008, 26, 901-912.

[ZEN18] Zeng, R.; Huang, H.; Yang, L. & Zhang, Z., “Joint Estimation of Frequency Offset and Doppler Shift in High Mobility Environments Based on Orthogonal Angle Domain Subspace Projection”, in *IEEE Transactions on Vehicular Technology*, 2018, 67, 2254-2266.

[ZHA20] Zhao, Y.; Niemegeers, I. G. & De Groot, S. H., “Distributed mmWave Massive MIMO: A Performance Comparison with a Centralized Architecture for Various Degrees of Hybridization”, in *IEEE International Conference on Information, Communication and Networks (ICICN)*, 2020, 105-110.