UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Doctoral Thesis

Performance Improvements for Multi-Radio Dual Connectivity in Mobile Networks

by:

Carlos Hernán Pupiales Yépez

Ph.D. Advisor:

Dr. Ilker Demirkol

Thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Network Engineering

Departament of Network Engineering

October 10, 2022

Abstract

The initial mobile network technologies were conceived to support mobile telephony; however, current mobile technologies are focused on supporting a diverse range of services such as interactive multimedia services, which demand stringent requirements. The fifth-generation mobile communication standard (5G) promises benefits on a wide range of new services generally categorized as ultra-reliable low-latency communications (URLLC), enhanced mobile broadband (eMBB), and massive machine-type communications (mMTC) [1]. For instance, 5G can provide faster data rates with lower packet latency that can facilitate the experience of new immersive technologies such as Virtual Reality (VR) and Augmented Reality (AR). The great advantage of the new 5G technology over its predecessors lies in the potential it has to transform final consumers' lives and how businesses run. For example, 5G promises to meet some extreme industrial requirements such as low latency and high data reliability, which can drive to cost savings and increase the efficiency of various industrial processes. Indeed, the manufacturing sector can expect productivity gains of around 20-30% [2]. In this regard, it is forecasted that 5G will generate $3.8 trillion of gross output and will support a value chain of 22.8 million new jobs by 2035 [3].

Initial 5G networks were deployed to satisfy the users with higher data rates. Hence, to reduce the time to market and maximize the value of already deployed infrastructure, such deployments used the Non-Standalone architecture, which involves a complex coupling with 4G networks using the Multi-Radio Dual Connectivity (MR-DC) technology. This technology allows the user equipment (UE) to maintain a simultaneous connection with two base stations (BSs), i.e., 4G and 5G BSs. Hence, the UE in MR-DC operation can achieve higher data rates by utilizing both BSs for a given data stream. In principle, such a combination may result in data rate improvements. However, the heterogeneous radio link conditions experienced between the UE and each BS, the variable bandwidth resources assigned to the UE at each BS, and the non-zero delay backhaul connection between BSs may cause out-of-order packet reception, which can negatively affect the performance of transport layer protocols, especially for TCP.

This thesis, first, delves into the problem of data aggregation in MR-DC operation and, secondly, proposes solutions that allow the UE to achieve, or at least approximate, the theoretical aggregate throughput in static and mobile scenarios without diminishing data reliability and packet latency targets. In this regard, we first evaluate as realistically as possible the performance of TCP and UDP protocols in a Dual Connectivity testbed, which is implemented using the LTE/NR compliant Open Air Interface software and Software Defined Radios. Through our studies, we showed how the flow control and packet reordering decisions, done at the Packet Data Convergence Protocol (PDCP) layer, impact the performance of the aforesaid transport layer protocols. Specifically, we discovered that the achievable aggregate throughput depends

on the timeout value chosen to wait for any delayed PDCP PDU and how the PDCP PDUs are shared between the BSs. Additionally, we showed that, unlike TCP, the out-of-order reception of PDUs is not a significant concern for UDP from the transport layer perspective. However, from the application layer's point of view, the out-of-order packet reception can be treated as packet loss.

Based on the outcomes described above, we focused on developing an efficient flow control mechanism, which, unlike other state-of-the-art flow control algorithms, approximates the ideal aggregate throughput for TCP and UDP transport layer protocols. Such a performance is achieved by dynamically splitting the incoming user traffic based on the assigned bandwidth resources and buffering delay experienced at each BS. The admirable feature of this novel mechanism is that it approximates the ideal aggregate throughput independently from the value chosen for the PDCP reordering timeout. Hence, this feature significantly simplifies the implementation and operation complexities of the proposed solution on production networks. We showcased the advantages of the proposed solution by benchmarking it against other state-of-the-art flow control algorithms under realistic scenarios using an LTE/NR testbed.

In the last part of this thesis, we studied the case where the UE aggregates data and handovers/failure events occur at the BSs. Results revealed that the UE's temporal disconnection from the BS can cause out-of-order reception or loss of PDCP PDUs, which makes the application stop receiving data for several hundreds of milliseconds. This problem challenges meeting the reliability and throughput targets defined by the application. Since there is no solution to overcome this problem in the available literature, we introduced a novel mechanism that gives the BS acting as the master node the capability to identify and timely retransmit the missing PDCP PDUs. Our experimental evaluations demonstrated that the proposed solution achieves near-zero data interruption time at the application level and data reliability of more than 99.999% without transport layer retransmissions in all the assessed scenarios.

Acknowledgements

This thesis does not just reflect the culmination of my research stage at the Universitat Politècnica de Catalunya but the enormous effort I put every single day into pushing my limits, overcoming my fears, and rediscovering myself. I want to thank all the lovely people who were with me during this stage. First of all, I want to thank my family for their infinite love and support, especially my parents, Marina and Carlos.

I also want to thank the nice guys I met at the UPC, Juan, Leticia, Godfrey, JB, Christian, Akshay, Pablo, Byron, Victor, and Joseph, which whom I spent deep conversations, hours of laughs and complaints, hundreds of coffees, and unforgettable moments.

Thanks also to my friends in Ibarra, Harold, Carmelo, and Lito, with whom the distance was not a problem to be in touch. Thanks for your motivational words and for trusting in me, even when I did not trust myself. Thanks also to my lovely friends in Sants, Carlos and Vai.

Sincere thanks to my doctoral thesis advisor Dr. Ilker Demirkol, for guiding me with infinite patience, understanding the challenges I was facing, and becoming a friend instead of an advisor only. I am sure I had the best advisor ever.

Finally, I want to thank Secretaría Nacional de Educación Superior Ciencia y Tecnología (SENESCYT) for providing funding to pursue doctoral studies through the scholarship program "Programa de Becas Internacionales de Posgrado 2018".

Contents

List of Figures

List of Tables

Acronyms

5G	fifth-generation mobile communication standard
VR	Virtual Reality
AR	Augmented Reality
KPI	Key Performance Indicator
QoS	Quality of Service
BS	Base Station
CA	Carrier Aggregation
UE	User Equipment
RAT	Radio Access Technology
mMIMO	Massive multiple-input multiple-output
SINR	Signal-to-interference-plus-noise ratio
MC	Multi-Connectivity
MNO	Mobile Network Operator
LTE	Long Term Evolution
NR	5G New Radio
URLLC	Ultra Reliable Low Latency Communications
eMBB	enhance Mobile Broadband
mMTC	Massive Machine Type Communications
CN	Core Network
RAN	Radio Access Network
3GPP	3rd Generation Partnership Project
DC	Dual Connectivity
MR-DC	Multi-Radio Dual Connectivity
PDCP	Packet Data Convergence Protocol
PDU	Packet Data Unit
RRM	Radio Resource Management
TCP	Transmission Control Protocol
RLC	Radio Link Control
RF2	Frequencies in the range 2
HO	Handover
RLF	Radio Link Failure
UDP	User Datagram Protocol
MPTCP	Multipath TCP
OAI	Open Air Interface
CCW	Capacity and Congestion Aware
MN	Master Node
SN	Secondary Node
MAC	Medium Access Control
SDU	Service Data Unit

SC	Single Connectivity
UP	User Plane
CQI	Channel Quality Indicator
FaRe	Fast Data Recovery
CP	Control Plane
RRC	Radio Resource Control
PHY	Physical
multi-TRP	Multi-Transmission and Reception Points
CoMP	Coordinated Multi-point
TB	Transport Block
HARQ	Hybrid Automatic Repeat Request
NR-U	5G NR in Unlicensed Spectrum
LAA	License-assisted Access
DAPS	Dual Active Protocol Stack
RA	Random Access
LWA	LTE-WLAN Aggregation
IP	Internet Protocol
LWIP	LTE WLAN Radio Level Integration with IPsec Tunnel
WLAN	Wireless Local Area Network
ATSSS	Access Traffic Steering, Switching, and Splitting
HO	Handover
MPQUIC	Multipath QUIC
V2X	Vehicle to Everything
EPC	Evolved Packet Core
5GC	5G Core Network
EN-DC	E-UTRA-NR Dual Connectivity
NE-DC	NR-E-UTRA Dual Connectivity
NGEN-DC	Next Generation-RAN E-UTRA-NR Dual Connectivity
SRB	Signaling Radio Bearer
DRB	Data Radio Bearer
S-SN	Serving Secondary Node
T-SN	Target Secondary Node
SDR	Software Defined Radio
TTI	Transmission Time Interval
CSI	Channel State Information
MCS	Modulation and Coding Scheme
BLER	Block Error Rate
PRB	Physical Resource Block
TBS	Transport Block Size
EWMA	Exponentially Weighted Moving Average
CC_R	Capacity and Congestion Report
DDDS	Downlink Data Delivery Status
FIFO	First-In First-Out
FR2	Frequencies in the Range 2
RSRP	Reference Signal Received Power

Dedicated to my lovely family,
we did it!

Chapter 1

Introduction

Every new generation of mobile networks is expected to support a broader range of services with distinct key performance indicators (KPIs). For instance, 5G critical services require very low latency and a high level of reliability, whereas broadband services demand support for high traffic density per unit area and high data rates per user [1]. While supporting such diverse quality of service (QoS) requirements, it is critical to use the spectrum resources efficiently. For instance, several approaches to improve the user data rate can be considered, such as increasing the bandwidth, improving the spectral efficiency, implementing cell densification, and enabling effective coordination between multiple base stations (BSs) [4]. Nevertheless, spectrum resources are scarce and costly, hardware complexity and its cost limit spectral efficiency, and cell densification results in higher interference along with higher capital and operating expenditures. Technologies such as Carrier Aggregation (CA), where the user equipment (UE) consumes radio resources of the same BS and the same radio access technology (RAT), has been widely used to improve the user throughput. However, it is still limited by the scarcity of the bandwidth resources assigned to a BS. Likewise, beamforming and massive multiple-input multiple-output (mMIMO) systems can improve the signal-to-interference-plus-noise ratio (SINR) and thus, increasing the obtained data rate. Nevertheless, high manufacturing costs and hardware complexities still limit their wide adoption.

Multi-connectivity (MC) emerges as an alternative solution that allows a mobile network operator (MNO) to leverage bandwidth resources from different BSs to enhance user performance. MC allows the UE to simultaneously consume radio resources of multiple BSs operating the same or different RATs such as Long Term Evolution (LTE), 5G New Radio (NR), and Wi-Fi. In this regard, MC can play an essential role in achieving several KPIs targets defined for the typical 5G use case scenarios: URLLC, eMBB, and massive machine-type communications mMTC [1]. For instance, MC can flexibly attain improved data rate through data aggregation for eMBB, improved data reliability through path redundancy for URLLC, or mobility robustness through data offloading for eMBB or URLLC. Such objectives can be achieved at reasonable implementation costs, but different MC designs may entail different levels of coordination, and hence different deployment and operation complexity.

In any MC design, the user plane (UP) traffic can be split, duplicated, or offloaded at a given transmitting protocol layer and then reversely aggregated or assembled in the mirrored receiving protocol layer. The splitting, duplication, or offloading is possible since the aforementioned protocol layer, named MC anchor layer in this thesis, and the layers above serve as a common entity for independent communication paths formed by the lower layer protocol stacks. Different MC architecture options can be envisioned to enable MC operation, wherein the MC anchor layer

can be located at the core network (CN), radio access network (RAN), or application server (end-to-end). Likewise, different technologies can be considered MC solutions depending on which protocol layer serves as the MC anchor layer. For instance, the 3rd Generation Partnership Project (3GPP) has standardized the Dual Connectivity (DC) [5] technology, wherein the MC anchor layer is located at the Packet Data Convergence Protocol (PDCP) layer.

DC has gained significant attention among the possible MC solutions since it allows the UE to communicate with two BSs simultaneously. Hence, this feature made possible the first deployments of 5G networks in a non-standalone fashion, in which the NR's BS, i.e., the gNB, is connected to the CN via the LTE's BS, i.e., the eNB [6]. For this, the 3GPP has extended DC to Multi-Radio Dual Connectivity (MR-DC) [6], wherein the UE can communicate with two different RATs, i.e., LTE and NR, as shown in Fig. 1.1. For MR-DC, data aggregation, for downlink traffic, implies aggregating PDCP Packet Data Units (PDUs) that are transmitted to the UE via two BSs, which manage their own Radio Resource Management (RRM) procedures. In addition, these BSs are geographically separated but logically connected via a backhaul with a non-zero delay [4, 7]. Because of this, the transmitted PDUs will arrive at the receiver's anchor layer at distinct time instances. Indeed, the delay that each PDU experiences at the corresponding communication path, i.e., the sojourn time, is significantly influenced by the assigned bandwidth resources and mobility patterns experienced between the UE and the corresponding BS.

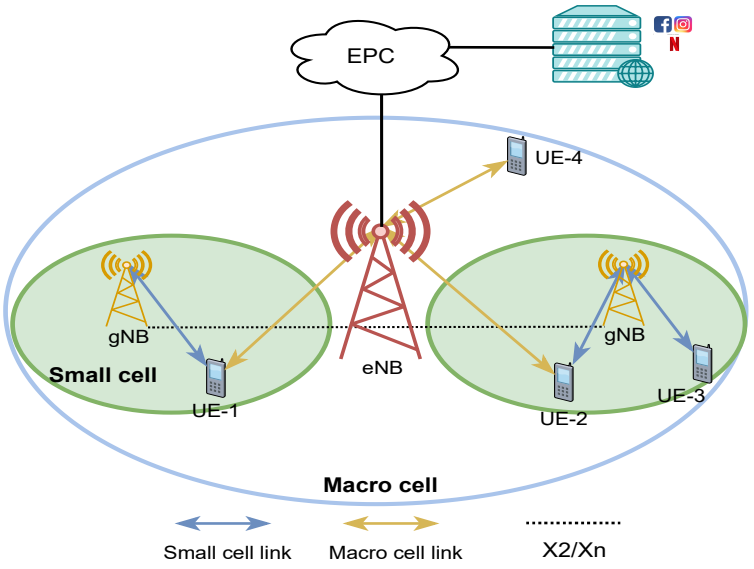


FIGURE 1.1: Initial MR-DC deployment scenario.

The delay difference mentioned above causes the PDUs to arrive out of order at the receiving PDCP layer. However, when a PDCP PDU sequence number gap is detected, all the subsequent received PDUs are buffered in the receiver PDCP layer until the sequence gap is filled or a reordering timer expires [8]. If a delayed PDU is not received before the reordering timer expires, the data with sequence gaps are delivered to the upper layers. Having a sizeable waiting time can avoid the aforementioned problem. Nonetheless, this can also affect the performance of latency-constrained applications and those relying on the Transmission Control Protocol (TCP). Indeed, the excessive waiting time at the receiver PDCP layer can make the server TCP retransmit the non-acknowledged data, i.e., to perform retransmission by timeout, which even increases the period that the application does not receive data.

In the typical MR-DC deployments, as depicted in Fig. 1.1, the backhaul delay and the Radio Link Control (RLC) buffering delay are the main contributors to the sojourn time difference mentioned above. The former introduces a few milliseconds of one-way latency. Whereas the latter significantly varies depending on how much data the BSs can transmit at every transmission opportunity. For instance, if the channel capacity is low compared to the previously assigned rate and assuming that the incoming data keeps piling up at the same speed, the buffered data may temporarily increase, and thus, the RLC buffering delay. In light of this, the out-of-order packet arrivals and the temporarily varying radio conditions challenge achieving a high and stable aggregate throughput. To avoid this problem, the incoming user traffic should be distributed via both BSs in such a way the RLC buffers have sufficient data to be transmitted at each transmission opportunity but without increasing the buffering delay [9]. Unfortunately, the heterogeneity of network deployments, radio link conditions, mobility patterns, and QoS requirements challenge having such an effective flow control logic.

Furthermore, the use of frequencies in the range 2 (RF2), i.e., from 24.25 GHz to 52.6 GHz [10], in one or both BSs can increase the likelihood of handovers (HOs), and it can lead to frequent radio link failures (RLFs). The former results from the mobility through smaller cell coverage areas, while the latter is the consequence of higher susceptibility to signal blockages caused by objects such as people, cars, and buildings. When the UE aggregates data, and the aforementioned events occur in one of the BSs, the communication between the UE and the corresponding BS is no longer available, causing out-of-order reception or losses of PDCP PDUs. Unfortunately, the time to complete the HO procedure and recover from an RLF is not negligible. Hence, the data interruption time experienced by the application, which can range from dozens to hundreds of milliseconds, can seriously challenge meeting the throughput, latency, and reliability targets defined for some emerging 5G services, such as the low-latency eMBB services.

1.1 Objectives and Contributions

The main goal of this thesis is to propose solutions to efficiently aggregate traffic from two 3GPP BSs using Multi-Radio Dual Connectivity. Specifically, we aim to maximize the aggregate throughput for User Datagram Protocol (UDP)- and TCP-based traffic in static and mobile scenarios. In this regard, the aim of this thesis has been accomplished through the following contributions:

- *Understanding Multi-connectivity*: Satisfying the stringent 5G QoS requirements necessitates efficient resource utilization by mobile networks. Consequently, we argue that Multi-connectivity is an effective solution to leverage the limited radio resources from multiple BSs in order to enhance user throughput, provide seamless connectivity, and increase data reliability. For this, we study different MC architectures, where distinct network entities and protocol layers are used to split or aggregate the user traffic. The benefits and challenges of MC are analyzed qualitatively as well as the open issues that network/device vendors and MNOs have to address for its use. Finally, through experimental evaluations, we illustrate the importance of MC design decisions for overall network performance quantitatively. The relevant study is published in *IEEE Communications Magazine* [**J2**].
- *Addressing the data aggregation problem*: The key aspect of traditional data aggregation protocols such as Multipath TCP (MPTCP) is the ability to control the end-to-end congestion in each communication link independently. Nevertheless, in MR-DC, the splitting

and aggregation processes are executed at the PDCP layer, making it impossible for the transport layer protocols to efficiently respond to congestion events and/or packet losses that arise in the communication links. Therefore, these protocols are inefficient to enable data aggregation in MR-DC operation. In addition, typical LTE/NR simulators and emulators lack a complete protocol stack implementation, i.e., they do not fully implement the transport and upper layers functionalities, maybe limiting the comprehension of the impact of MR-DC on the performance of transport layer protocols. For this reason, we implemented MR-DC on a novel LTE/NR testbed using the Open Air Interface (OAI) software and commodity hardware. Consequently, we studied how TCP- and UDP-based traffic behaves in a real multi-RAT DC scenario. The results of this study were published in *16th IEEE International Conference on Mobile Ad-Hoc and Smart Systems (MASS)* [**C1**], and in *IEEE Communications Magazine* [**J2**]. Lastly, it is worth mentioning that the work [**C1**] was awarded the Best Demo Award at the IEEE MASS conference in December 2019.

- *Designing a network-adaptive method for traffic splitting*: Each BS independently manages the assignment of radio resources to the UE in MR-DC operation depending on experienced radio link conditions, available radio resources, and QoS requirements. Therefore, the aggregate throughput would equal the sum of the throughputs achieved at each BS when employing single connectivity (SC) operation. However, the variability that may occur in the aforementioned conditions, the random signal interruptions, and traffic fluctuations can cause under-utilized links or high buffering delays, decreasing the obtained aggregate throughput. To address this challenge, we designed and evaluated the Capacity and Congestion Aware (CCW) flow control algorithm that efficiently and dynamically splits the incoming user's traffic via the master node (MN) and secondary node (SN) based on their Medium Access Control (MAC) Service Data Unit (SDU) sizes and RLC buffering delay statistics. Our CCW allows the UE to approximate the theoretical aggregate throughput for any MAC scheduler design, MR-DC architecture option, and transport layer protocols. The results were published in the *IEEE Access Journal* [**J1**].

- *Dimensioning the reordering timeout value*: The use of a non-zero delay backhaul connection between the BSs along with the RLC buffering delay, created by the variability of the radio link conditions and assigned radio resources, cause the PDUs traversing the MN to experience different sojourn times compared to the PDUs traversing the SN. For this reason, a reordering timer is used, at the receiver, to wait for the delayed PDUs. Nonetheless, the heterogeneity of network deployments, user mobility patterns, and radio link conditions challenge having a single reordering timeout value to work efficiently in all the above scenarios. Indeed, the excessive, or insufficient, waiting time can cause bufferbloat, or spurious PDU discards, respectively. We tackle this problem by intelligently limiting the RLC buffering delay at both BSs. Hence, the reordering mechanism can use a single and small reordering timeout value. This functionality has been included as part of our CCW flow control algorithm, and the results were published in the *IEEE Access Journal* [**J1**].

- *Minimizing the data interruption time at the higher layers*: In a typical MR-DC deployment, the mobility events and the radio link failures, which may occur in the SN during the data aggregation, can cause data interruption periods that, in turn, create out-of-order deliveries or data losses at the receiver PDCP layer. Because of these abnormalities, the transport and application layers stop receiving data for up to hundreds of milliseconds, which challenges meeting the KPIs defined for many 5G use cases. To address this problem, we have proposed an intelligent and efficient mechanism that timely retransmits via

the MN the non-delivered/transmitted PDUs. Using LTE/NR testbed experiments, we demonstrate that the proposed mechanism achieves near-zero interruption time and data reliability of more than five nines, i.e., 99.999%, without transport layer retransmissions for saturated TCP traffic. The obtained results were published in the *IEEE Access Journal* [**J3**].

1.2 Resulting publications

Most of the contents of this dissertation have been published in the following journals and conferences:

JCR Journal Publications:

J1 C. Pupiales, D. Laselva, and I. Demirkol, "Capacity and Congestion Aware Flow Control Mechanism for Efficient Traffic Aggregation in Multi-Radio Dual Connectivity," in IEEE Access, vol. 9, pp. 114 929–114 944, August 2021 [9]. (Area: Telecommunications; Rank: 43/94; Quartile: Q2; IF: 3.476).

J2 C. Pupiales, D. Laselva, Q. De Coninck, A. Jain, and I. Demirkol, "Multi-Connectivity in Mobile Networks: Challenges and Benefits," in IEEE Communications Magazine, vol. 59, no. 11, pp. 116–122, November 2021 [11]. (Area: Telecommunications; Rank: 7/94; Quartile: Q1; IF: 9.030).

J3 C. Pupiales, D. Laselva, and I. Demirkol, "Fast Data Recovery Mechanism for Improved Mobility Support in Multi-Radio Dual Connectivity," in IEEE Access, vol. 10, pp. 93 674–93 691, September 2021 [12]. (Area: Telecommunications; Rank: 43/94; Quartile: Q2; IF: 3.476).

Refereed Conferences:

C1 C. Pupiales, W. Nitzold, C. Felber, and I. Demirkol, "Software-based Implementation of Dual Connectivity for LTE," in Proc. IEEE 16th Int. Conf. Mobile Ad Hoc Sensor Syst. Workshops (MASSW), Nov. 2019, pp. 178–179 [13].

International Workshops:

The following presentation contributed to the development of the OpenAirInterface project.

W1 C. Pupiales and I. Demirkol, "Efficient Traffic Aggregation for Dual Connectivity," in OAI Virtual Workshop 2021, Jun 2021 [14].

1.3 Outline of this Thesis

Once we have described the motivation and objectives of this thesis, the following chapters of this document are organized as follows.

In Chapter 2, we introduce the general concept of multi-connectivity and its different architecture options, which can improve user throughput, increase data reliability, and provide seamless connectivity. We detail the advantages provided by MC, along with the open research issues that are crucial to be studied to achieve an efficient MC operation. In addition, we review the standardized MC solutions and compare their benefits and limitations qualitatively. We explain in detail the 3GPP-standardized MR-DC solution. Furthermore, to show the challenges that MC operation brings to overall system performance, we showcase the impact of MC design decisions on the end-to-end system performance using an LTE/NR-compliant testbed.

In Chapter 3, we study the data aggregation problem in MR-DC from the UP perspective. Then, we present a novel flow control algorithm named Capacity and Congestion Aware (CCW) Flow Control, which efficiently aggregates TCP- and UDP-based traffic based on the assigned radio resources and buffering delay statistics of both BSs. In addition, we evaluate the performance of the proposed method against state-of-the-art and benchmarking flow control algorithms using an LTE/NR-complaint testbed, which uses a real radio link Channel Quality Indicator (CQI) dataset.

Furthermore, Chapter 4 discusses how the mobility events and the RLFs, which may occur in one of the BSs during the data aggregation, pose challenges for MR-DC in meeting the KPIs defined for many 5G use cases. To tackle this challenge, we present an intelligent and efficient mechanism named Fast Data Recovery (FaRe) that significantly minimizes the application's data interruption time experienced during HOs or RLFs in the SN link. Lastly, using an LTE/NR testbed, the performance of the FaRe is evaluated against different benchmarking solutions.

Finally, we summarize the conclusions of this thesis in Chapter 5. Likewise, the chapter presents the future work guidelines to help MR-DC become a more efficient MC solution.

Chapter 2

Background on Multi-Connectivity

In this chapter, we describe Multi-connectivity in detail. For this, we introduce its different architecture options, the advantages MC provides, and the open research questions that are significant to achieving efficient MC operation. Moreover, we explain the diverse standardized MC solutions that can be enabled by placing the MC anchor layer at different protocol layers. At this point, we put more effort into describing the 3GPP MR-DC solution. Lastly, we showcase the impact of MC design decisions and their importance on end-to-end system performance using an LTE/NR testbed.

Contributions:

- C. Pupiales, W. Nitzold, C. Felber, and I. Demirkol, "Software-based Implementation of Dual Connectivity for LTE," in Proc. IEEE 16th Int. Conf. Mobile Ad Hoc Sensor Syst. Workshops (MASSW), Nov. 2019, pp. 178–179.
- C. Pupiales, D. Laselva, Q. De Coninck, A. Jain, and I. Demirkol, "Multi-Connectivity in Mobile Networks: Challenges and Benefits," in IEEE Communications Magazine, vol. 59, no. 11, pp. 116–122, nov 2021 (Area: Telecommunications; Rank: 7/94; Quartile: Q1; IF: 9.030).

2.1 Multi-Connectivity Architectures

Splitting, duplicating, or offloading the user traffic requires to have a common protocol layer, i.e., the MC anchor layer, for multiple protocol stacks. In this regard, different MC architecture options can be envisioned depending on where the MC anchor layer is located, i.e., at the CN, RAN, or application server. In the following, we present such architectures, along with several MC standardized solutions.

2.1.1 RAN-based MC

In RAN-based MC, radio resources from multiple radio paths are coordinated among multiple BSs for a given UE. Depending on the MC solution, BSs might communicate using a specific interface through the backhaul, e.g., 5G BSs communicate through the Xn interface [15]. In contrast, if at least one LTE BS is in use, the X2-U interface is used [16]. In this case, one

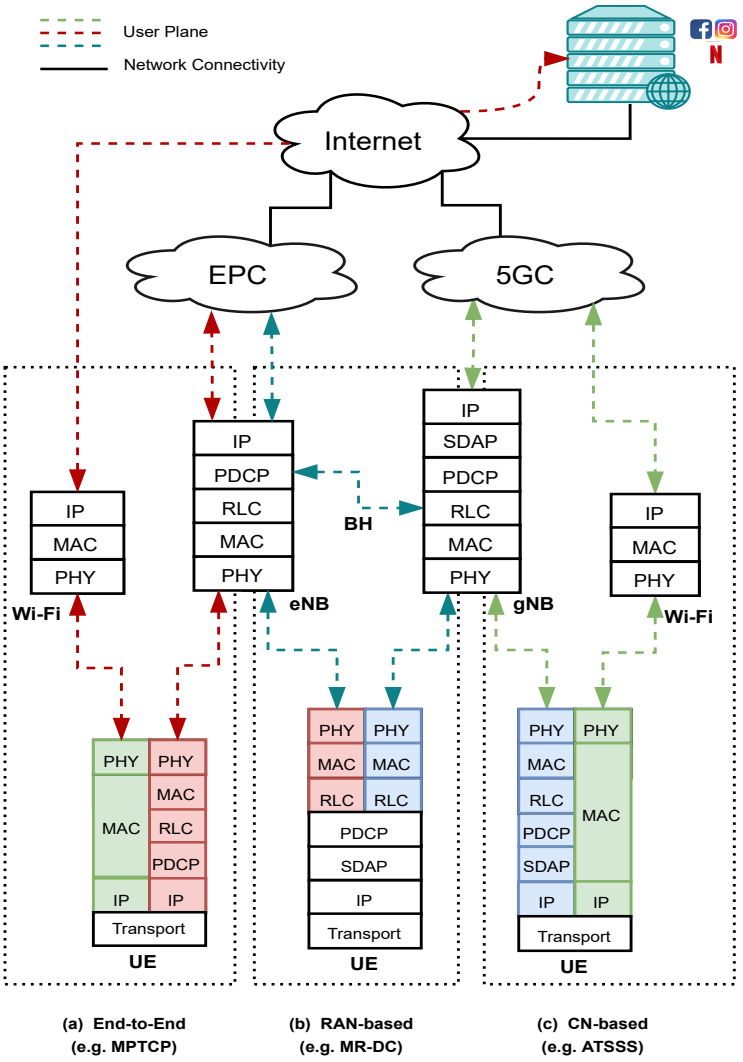


FIGURE 2.1: MC architecture examples from the user plane perspective.

of the BSs acts as an anchor entity to host the MC anchor layer for the UP and to manage the MC-related control plane (CP) procedures regardless of the CN configuration. Fig. 2.1b illustrates the UP implementation for a RAN-based MC solution.

The anchor BS uses the Radio Resource Control (RRC) protocol to exchange signaling messages among the BSs to maintain the MC operation with the UE. Note that radio-specific functionalities for each communication path are handled independently by each BS. Additionally, having the MC anchor layer at the RAN helps RRM algorithms quickly respond to sudden channel quality changes. However, the hardware and/or software upgrade costs required for every BS may limit the scalability of MC deployments. Although the communication between the anchor layer and the communication paths located on different BSs seems to be similar to that of functional splits [17], e.g., F1 interface defined by 3GPP, the messaging and context information required for MC are different. However, similar to the functional split options, the communication between the entities mentioned above might define latency and/or capacity requirements for the backhaul that may challenge the effectiveness of MC solutions. In the following, possible MC anchor layers for the RAN-based MC architecture option are explained.

2.1.1.1 Physical Layer

In the Physical (PHY) MC, PHY-MC, the use of spectrum and/or time resources from multiple BSs are coordinated for a given UE. The UE's interference information is used to decide the BSs to communicate with the UE at a given time. For PHY-MC to be efficient, a continuous control information exchange is necessary between the UE and BSs, and among BSs. Such a level of coordination defines stringent latency requirements, especially for backhaul links. Moreover, complex solutions are required to support different numerologies, frame structures, and waveforms potentially used by different BSs. Fig. 2.2 shows an example of the signaling exchange between different BSs required to support PHY-MC solutions.

Multi-Transmission and Reception Points (multi-TRP), in 5G, and Coordinated Multi-point (CoMP), in 4G, are MC technologies, which coordinate the transmission and reception of user data from several BSs in order to mitigate the inter-cell interference. Unfortunately, the above mentioned challenges limit the effective adoption of such technologies [18, 19].

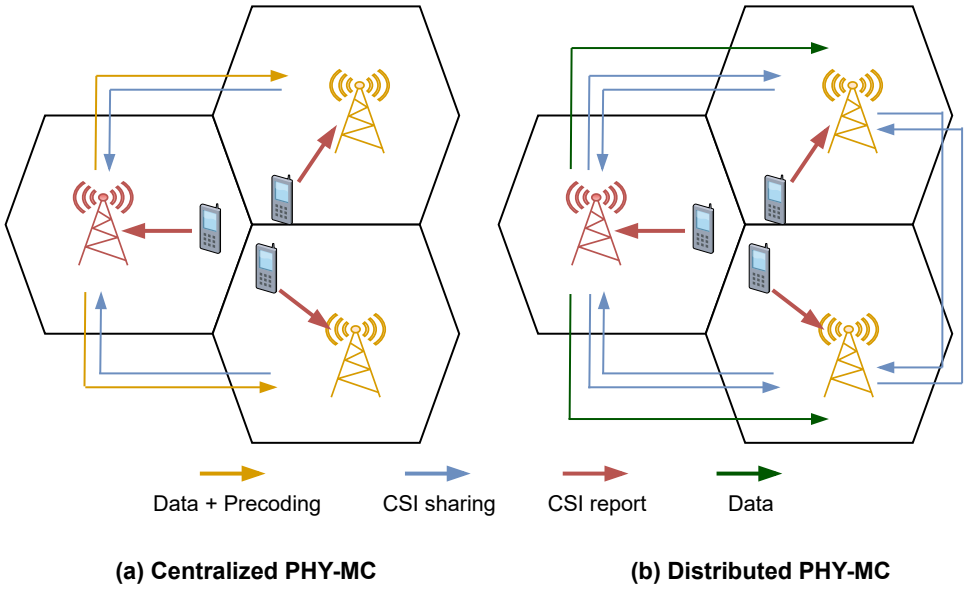


FIGURE 2.2: Signaling exchange for the centralized and distributed implementation schemes for the Joint Transmission CoMP [19].

2.1.1.2 Medium Access Control Layer

In MAC-MC, the transmission of transport blocks (TBs) is coordinated with multiple PHYs. The main benefit of MAC-MC is the fast traffic switching between communication paths thanks to the rapid system adaptation in case of radio link changes. Nevertheless, supporting a single MAC entity with different transport block sizes, time slot duration, and Hybrid Automatic Repeat Request (HARQ) procedures defined for the RATs will considerably increase the scheduler complexity. For instance, NR uses a codeblock-based asynchronous HARQ for uplink and downlink, while LTE uses TB-based synchronous HARQ for uplink and asynchronous for downlink. In addition, the signaling overhead required to differentiate the TBs from different RATs correctly may also increase. Furthermore, when the PHY and MAC reside in different locations, e.g., Central Unit - Distributed Unit, the delay added in the backhaul connection challenges

achieving the smaller slot durations envisioned for URLLC applications. Note that this scenario can be implemented using the option 6 of the functional splits architecture [17].

5G NR in Unlicensed Spectrum (NR-U) with CA can be considered a MAC-MC solution since it uses a license-assisted access (LAA) method to transmit user traffic via different RATs located at the same BS, using licensed and unlicensed spectrum, respectively [20]. In this case, the MAC from the NR stack controls or coordinates both communication paths.

2.1.1.3 Packet Data Control Protocol Layer

PDCP is suitable to act as an MC anchor layer since its procedures do not face very tight timing constraints as MAC or PHY do and because its specifications are similar for LTE and NR. Hence, each communication path can perform its link adaptation and resource allocation procedures. In this sense, it is possible to increase the user throughput or the data reliability by splitting/aggregating or duplicating a PDU using multiple BSs without incurring major implementation complexities [21]. The following 3GPP technologies can be considered as PDCP-MC solutions:

- **Dual Connectivity and Multi-Radio Dual Connectivity:** The UE is simultaneously served by two 3GPP BSs, with the same RAT for the former and different RATs for the latter solution. In both cases, the MC-related CP aspects are handled by one of the two BSs, while the UP uses one or both BSs simultaneously [4, 7].
- **NR-U with DC:** The UE can communicate simultaneously with two 3GPP BSs, the RATs of which operate on licensed and unlicensed spectrum, respectively [20]. The anchor BS, either LTE or NR, manages the NR-U BS and the MC user and control planes.
- **NR Dual Active Protocol Stack (DAPS):** The UE is simultaneously connected to two NR BSs to ensure seamless handover. In this regard, the UE maintains the communication with a source BS until the Random Access (RA) procedure with the target BS is completed. The UE releases the connection with the source BS after receiving an explicit notification from the target BS. Unlike other 3GPP-defined DC solutions, in DAPS, there is no anchor BS.
- **LTE-WLAN Aggregation (LWA):** It integrates LTE and Wi-Fi networks controlled by the same MNO. The eNB decides whether to offload the traffic to Wi-Fi or aggregate it between the LTE and Wi-Fi networks. The 3GPP defines a new EtherType and an LWA Adaptation Protocol to avoid changes to the WLAN MAC layer. The former allows the UE to differentiate between LWA and non-LWA packets, while the latter serves for bearer identification [5].

2.1.1.4 Network Layer

The network layer can act as the MC anchor layer since the IP protocol is the de facto network layer protocol for all 3GPP and non-3GPP networks. The MC operation between 3GPP and non-3GPP BSs requires an IP tunnel between the UE and the 3GPP BS for security and encapsulation purposes. Moreover, different IP addresses should be configured at the UE to route the IP packets. However, MNOs usually configure a single IP address per UE in their CN. Therefore, new functionalities in the IP header may be required to support traffic aggregation,

e.g., packet sequence identification. These functionalities bring implementation complexities and additional costs that may limit the MC deployments.

LTE WLAN Radio Level Integration with IPsec Tunnel (LWIP) is a Network-MC solution, where data transfer from LTE to WLAN and vice versa is done using an IPSec tunnel. LWIP does not require any changes on the WLAN infrastructure, unlike the LWA. The eNB manages the MC-related CP and UP functionalities, while each BS handles its radio-specific functionalities.

2.1.2 CN-based MC

In this MC architecture option, the CN directly manages the MC-related CP and UP functionalities. For this, BSs that are connected to the same or different CNs can serve the UE, as depicted in Fig. 2.1c. The latter case requires coordination between both entities, e.g., between LTE and 5G CNs, for which new communication procedures are needed. Nevertheless, in such a scenario, a RAN-based MC approach results in a simpler alternative. In both cases, disjointed communications paths are required to transport the user traffic through multiple BSs. This can be done using independent CN connections for each BS, and thus, different IP addresses at the UE. Even though this approach is an affordable solution that offers versatility and scalability regardless of the number of BSs and their technology, it also imposes the following challenges for MNOs:

- Implementation of new capabilities to split and aggregate UP traffic that belongs to different packet data networks.
- Quickly adapting to the dynamic radio link conditions since the CN typically has no information about that.

In this regard, 3GPP introduced the *Access Traffic Steering, Switching, and Splitting (ATSSS)* technology that gives the MNO the control to steer UP traffic between NR and Wi-Fi networks. ATSSS uses specific multipath functions between the CN and UE to transport the user traffic and CP messages, such as traffic and round trip time measurements. Currently, ATSSS only supports TCP-based traffic with the use of the MPTCP protocol. Fig. 2.3 illustrates the architecture for ATSSS.

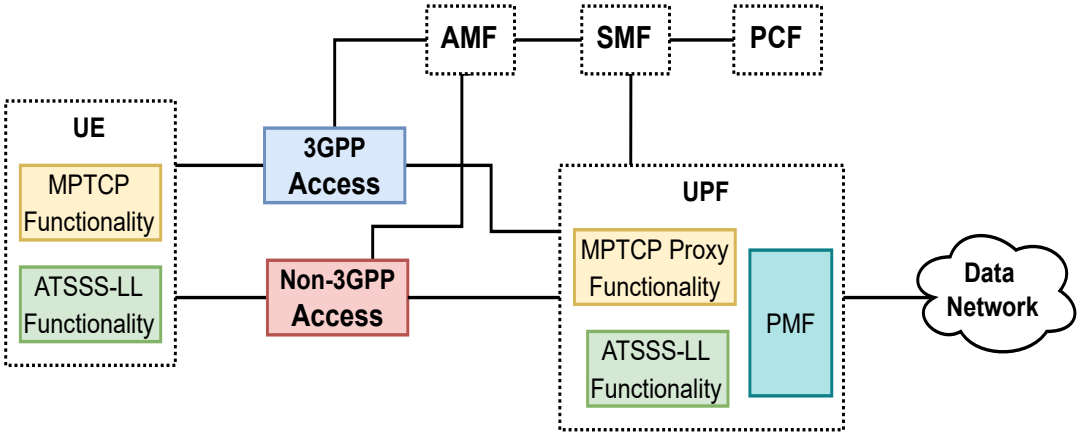


FIGURE 2.3: General architecture for ATSSS [22].

2.1.3 End-to-End MC

In this MC architecture option, MC operation is enabled using capabilities of upper-layer protocols between multiple 3GPP BSs or between 3GPP and non-3GPP BSs. In the former case, the UE and the CN can establish a redundant UP connection in an end-to-end fashion for reliability purposes [22], which is relevant for URLLC services. Nonetheless, it comes at higher deployment costs since it needs redundancy in all network entities, i.e., RAN and CN. In the latter case, MC operation is established between the UE and the application server. Hence it is entirely agnostic to the RAT and CN configuration. Indeed, this is the simplest approach to enable MC operation. However, it is not possible to guarantee a given QoS criteria since the MC operation is not in the control of the MNO. In both cases, *transport layer* protocols can be exploited to provide such functionalities. Fig. 2.1a shows the MC operation between LTE and Wi-Fi networks.

In this regard, MPTCP and Multipath QUIC (MPQUIC) protocols are affordable options to enable MC operation and transport TCP-based traffic. In this sense, independent traffic flows, one per communication path can be created using different IP addresses and/or port numbers [23], [24]. Note that MPTCP distributes a data stream into independent TCP flows across different IP addresses and/or port numbers, where each flow handles its congestion control. Similarly, MPQUIC uses application and transport layer functionalities, offering a faster connection establishment, multiplexing data streams, robustness against the head-of-line blocking problem, and packet encryption compared with MPTCP.

2.2 Benefits and Limitations

In this section, we describe the benefits that MC offers and the limitations encountered to provide an effective MC operation. The aforementioned aspects are summarized in Table 2.1 for the standardized MC solutions.

TABLE 2.1: Benefits and Limitations of MC Standardized Solutions

Anchor Layer	**Standardized Solution**	**Technical Specification**	**Technical Objective**	**Main Limitation**
PHY	CoMP	TS 36.300	Improvements in SINR	Requires high level of coordination between BSs
	Multi-TRP	TS 38.300		
MAC	NR-U CA	TS 38.331	Rapid system adaptation in case of link failures	Complex packet scheduler
PDCP	DC	TS 36.300	Higher data reliability; Higher throughput	Requires additional hardware and software capabilities at the UE
	MR-DC	TS 37.340		
	LWA	TS 36.300	Higher throughput	Aggregation limited to LTE and Wi-Fi controlled by the MNO
	NR-U DC	TS 38.331	Higher throughput	Requires additional hardware and software capabilities at the UE
	DAPS	TS 38.300	Mobility robustness	Currently available between 5G BSs
Network	LWIP	TS 36.300	Deployment affordability	Traffic offload limited to LTE and Wi-Fi
Transport (End-to-End)	MPTCP	RFC 6824	Higher throughput; Deployment affordability	MC operations are not in the control of the MNO
Transport (Core Network)	ATSSS	TS 24.193	Higher throughput; Higher data reliability	MC operation limited to TCP-based traffic

2.2.1 Improved Data Rate

In MC operation, the UE can combine multiple data streams from different BSs into a single data stream to enhance the user data rate. Ideally, the resulting aggregate throughput would equal the sum of the throughputs obtained at each communication path when operating SC. Nevertheless, the different radio link conditions and assigned radio resources experienced at each BS and the delay difference between both communication paths can cause out-of-order packet arrivals, negatively affecting the performance of the upper layer protocols, such as TCP. Indeed, the obtained MC throughput can even be lower than the one achieved in SC, as shown in [13]. Suppose a reordering mechanism is used to avoid this problem, as DC and MR-DC use. In that case, the maximum delay divergence is constrained by the reordering timeout value, i.e., dozens of ms. Additionally, the chosen MC solution might achieve application-level improvements depending on the network scenario. For instance, MPTCP and MPQUIC can increase the throughput only if they use disjointed communication paths. Likewise, the throughput gain would be evident when the channel capacities of the individual communication paths do not differ considerably.

2.2.2 Improved Reliability

In mobile networks, reliability is typically provided by retransmitting the erroneous data using retransmission protocols operating at the MAC and RLC layers. Despite being effective, it is also time-consuming. Hence, this approach is not suitable to simultaneously meet the reliability and low latency requirements needed by URLLC applications. In this regard, MC solutions can be exploited to provide reliability and low latency by sending redundant data using different BSs. For this, the Packet Duplication feature, defined for MR-DC, can satisfy the high reliability and low latency requirements, where identical control or data packets are sent through multiple BSs [21, 25].

2.2.3 Mobility Robustness

MC solutions can reduce the interruption time and the amount of signaling required for the seamless connection envisioned for future networks. For this, the UP traffic can be switched, i.e., offloaded, rapidly from one BS to another through the backhaul link, thus avoiding the intervention of the CN. For instance, the master BS provides the CP functionality to the UE in MR-DC. Hence, the UP is switched from the master to the secondary BS during a handover (HO) without involving the CN. This process is faster and offers lower signaling overhead compared to the traditional HO [4]. Even though one radio interface is sufficient for UP traffic, RRM procedures by both BSs are needed, so the UE must use both radio interfaces actively, like in the DAPS case. Similarly, MPTCP and MPQUIC can provide mobility robustness. However, in that case, the UE mobility is only possible between 3GPP and non-3GPP networks [23, 24]. It is worth mentioning that with the DAPS, MPTCP, or MPQUIC solutions, the user traffic is transmitted to the UE via a single BS. Hence, data aggregation is not possible during the handover.

2.2.4 Service Segregation

MC can be used to serve a UE requesting services with distinct requirements by segregating those services into different communication paths. For instance, the UE can be connected to two BSs whose RATs use mmWave and Sub-6 GHz carriers, respectively. The former can be used for services that demand high throughput, e.g., video streaming, while the latter can be used for lower throughput, such as mMTC. This feature may be suitable for emerging applications that define flows with different QoS criteria, such as Vehicle to Everything (V2X).

2.2.5 Deployment Cost Savings

In order to address the massive increase in throughput and the number of UE connections envisioned for eMBB applications, the MNOs have to extensively deploy 5G capabilities, especially in heavily populated areas. This deployment cost can be alleviated by adopting RAN-based MC solutions, like the one used for the 5G non-standalone mode, instead of massively deploying new costly RAN and CN infrastructures. This can be a cost-effective and scalable approach to reduce the time to market and bring the 5G capabilities to scenarios where completely deploying 5G infrastructure is not profitable, e.g., in rural areas. This strategy can also be used to satisfy temporary demands, e.g., for concerts or stadiums.

2.2.6 MC Messaging Overhead

For PHY-MC and MAC-MC solutions, the radio resource allocation at each time slot duration brings stringent requirements that increase the signaling traffic and limit the allowed backhaul delay. For instance, PHY-MC requires that the data for the UE is processed and forwarded to the corresponding BSs within one slot duration [19], e.g., 1 ms. For MAC-MC, the scheduling control and HARQ synchronization required by the MAC scheduler to serve multiple PHYs, limit the backhaul delay to 2 ms [17].

Contrarily, PDCP-, Network-, and CN-based MC require less frequent data usage reports, the frequency of which depends on the flow control method in use. Although these reports do not impose specific delay constraints for the backhaul, a considerable delay may affect the performance of transport and application layers, e.g., because of TCP's retransmissions by timeout. In general, how often MC control messages are exchanged should be decided based on the variability of the radio link conditions and target QoS. However, the overhead created by these messages increases linearly with the frequency they are exchanged, along with the number of UEs and BSs involved in the MC operation.

2.3 Challenges and Open Issues

Although MC offers several benefits as indicated in Section 2.2, there are several challenges and open issues that still need to be addressed. Table 2.2 summarizes the relevant open research questions that need further investigation.

Table 2.2: Summary of Open Research Questions

Packet Reordering	• How to handle packet reordering for a subset of the traffic when multiple traffic flows exist? • How long should a packet wait in the reordering buffer without affecting the communication performance?
Flow Control	• How to dynamically choose the splitting ratio considering the performance of the upper layer protocols?
Cross-layer Design	• How MC-based algorithms exploit the upper layer protocol information?
MC Operation Management	• Which is the entity in charge of managing the MC-related procedures? • What are the KPI targets to be considered for MC decisions?
Beyond Two Rats	• Is it possible to effectively aggregate data streams from more than two sources?

2.3.1 Packet Reordering

Different radio link conditions, RAT procedures, and communication path delays can cause packets to arrive out of order. Hence, a packet reordering solution is needed to avoid a negative impact on the performance of the upper layer protocols, e.g., for TCP. The 3GPP reordering method defined for DC and MR-DC [8] uses a static reordering timeout, the value of which must be carefully chosen. If not, packets can be excessively buffered at the receiver anchor layer, creating issues, such as bufferbloat, thus degrading the performance of time-sensitive applications. On the other hand, having a reordering timeout value that dynamically changes depending on how delayed the data is would be ideal. Nevertheless, this requires the receiver to know the buffering level and the flow control logic used at the transmitting side. Hence, the receiver can infer the expected delay for the data coming from different transmitters. A suitable reordering timeout value should be chosen considering aspects such as backhaul latency, radio link conditions, traffic type, buffer length, and QoS requirements.

2.3.2 Dynamic Flow Control

The flow control is the method used at the MN to split the incoming user traffic via the MN and SN. Hence, an inadequate flow control logic might create under- or over-utilized links, resulting in out-of-order packet arrivals and poor overall system performance. For instance, a static flow control logic is inefficient since the data rate from each BS is Spatio-temporal, i.e., it depends on the instantaneous radio link conditions and assignment of radio resources. If such aspects are not considered, it may be necessary to excessively buffer the PDUs at the sender and/or reorder them at the receiver, leading to inefficient use of the available radio resources. Factors such as the backhaul latency, radio link conditions, and QoS requirements rule the flow control logic.

2.3.3 Packet Duplication Optimization

Packet duplication implies independently transmitting the copies of the same PDCP PDU via both BSs. Hence, the receiver PDCP layer will only use the PDU that arrives first, discarding the other. In this regard, the redundant communication path should be aware of the successful reception of the PDU in order to avoid all subsequent MAC and/or RLC retransmissions that may occur in such a communication path. Note that these unnecessary retransmissions can delay the new PDUs to be received at the UE's PDCP layer.

2.3.4 Cross-layer Design

Flow control and reordering algorithms can make better decisions by considering the information from the different protocol layers, such as the application requirements, the transport protocol used, and the network conditions. For instance, unlike UDP, TCP has to ensure an in-sequence delivery, which might affect the application performance due to the delays and retransmissions it might incur. A careful design of the packet reordering at the receiving MC anchor layer can reduce such problems and improve the application performance. However, an inefficient MC reordering mechanism can result in additional delays that could cause spurious TCP retransmissions. Likewise, the flow control logic may need information from the lower layers of both BSs to split the incoming traffic accordingly.

2.3.5 MC Operation Management

The MNO should decide when to use MC instead of SC and which BSs should be involved in this MC operation. For instance, inefficient MC decision, user association, and resource allocation methods can degrade the performance of some UEs and even the overall system performance. For the decision-making, the MNO can consider the user QoS requirements, the terminal capabilities, and the spatio-temporal network KPIs. Such decisions can be improved by collecting and processing MC operation data through reinforcement learning and data analytics techniques.

2.3.6 Beyond Two RATs

Since recent UEs are already equipped with 4G, 5G, and Wi-Fi interfaces, they can use more than two BSs simultaneously to transfer the UP traffic. One of the benefits of this MC approach is the versatility of aggregating traffic even though one BS fails. This approach is not possible with the current MC standardized solutions since they consider only two BSs. Nevertheless, this new approach may increase the flow control and reordering algorithms' implementation and management complexity.

2.4 Multi-Radio Dual Connectivity Operation

In MR-DC operation, the UE is simultaneously connected to two BSs in a single RAT or heterogeneous RATs fashion. In the former, both BSs use either LTE or NR radio technology, and they are connected to their corresponding CN, i.e., Evolved Packet Core (EPC) for LTE and 5G Core (5GC) for NR. In the latter, the LTE's BS is connected to the EPC and NR's BS to the 5GC. In addition, the 3GPP defines different MR-DC architecture options depending on the technology used for the CN and anchor BS. For instance, in the E-UTRA-NR Dual Connectivity (EN-DC) option, the eNB and gNB are connected to the EPC. Fig. 2.4 depicts the currently supported MR-DC architecture options, where LTE-DC and NR-DC options are considered as single RAT. Moreover, NR-E-UTRA Dual Connectivity (NE-DC), Next Generation-RAN E-UTRA-NR Dual Connectivity (NGEN-DC), and EN-DC are considered as multi-RAT DC solutions [6].

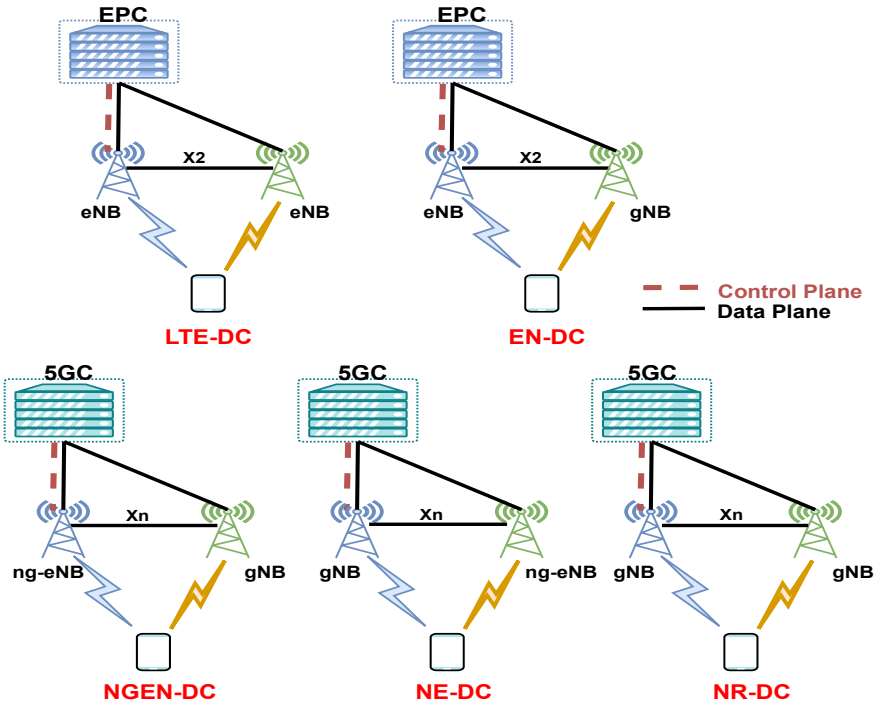


FIGURE 2.4: MR-DC architecture options.

2.4.1 Control Plane Aspects

In any MR-DC architecture option, the MN manages the control plane aspects between the UE and CN to support MR-DC operation. However, the MN and SN independently establishes an RRC connection with the UE. Hence, only the MN is responsible for managing all the CP-related procedures to initiate, maintain, and terminate the MR-DC operation between the UE and both BSs. Note that each BS handles its own RRM procedures through its own RRC protocol to initiate and/or maintain the connection between the BS and UE, such as random access and power control procedures [4].

From the radio bearer perspective, the MN uses two types of direct signaling radio bearers (SRBs), i.e., SRB1 and SRB2, to send RRC messages to the UE, thus supporting the MR-DC operation. On the contrary, the RRC messages from the SN, corresponding to the MR-DC operation, are not sent directly to the UE. Instead, they are forwarded to the MN, which transmits them to the UE. MR-DC also defines a new type of radio bearer, i.e., SRB3, which is used between the SN and UE to exchange information such as measurement reporting for mobility purposes within the SN's coverage area.

2.4.1.1 Downlink Data Delivery Status

In MR-DC operation, the downlink data delivery status (DDDS) procedure [26] helps the SN to provide feedback, via the X2/Xn interface, about the user data flow and the successful delivery of control data to the MN. In the case of RLC acknowledged mode (AM), the DDDS reports the highest PDCP PDU sequence number that is successfully delivered in sequence to the UE. Likewise, in the RLC unacknowledged mode (UM) case, the DDDS reports the highest PDCP PDU sequence number that is successfully transmitted to the UE. Moreover, the DDDS can

carry other information such as the desired buffer size in bytes for the concerned radio bearer, the sequence number of the packets declared as lost in the SN, and the retransmitted PDUs.

Note that the MN and SN will negotiate the frequency of the DDDS report exchange. However, the initial DDDS report is sent as soon as the UE completes the RA with the SN. Similarly, when the SN releases the split data radio bearer, the SN sends a DDDS report with the Final Frame Indication flag enabled. The MN can use the information provided by the SN in the DDDS report for flow control purposes. Hence, it may not be necessary to define new signaling messages for such a purpose.

2.4.2 User Plane Aspects

From the CN's perspective, UP traffic is transferred to/from the RAN using either the MN or SN, depending on which BS the PDCP layer resides. On the other hand, from the UE's perspective, the UP data can go through either one of the BSs or both BSs simultaneously, as illustrated in Fig. 2.5. This depends on the configuration, which can be on a per data radio bearer (DRB) level, and on the dynamic traffic aggregation decision [6]. In this regard, the UE can use DRBs belonging only to MN, SN, or both BSs. For first two cases, the 3GPP specifies the master cell group (MCG) and secondary cell group (SCG) bearers for MN and SN, respectively. For the latter, 3GPP defines the split DRB, which allows the UE to consume radio resources from both BSs at the same time [6].

Furthermore, in any MR-DC architecture option, the transmitting PDCP layer is in charge of splitting the user traffic between the available communication paths for increased throughput (aggregation), offloading the user traffic to one of the available communication paths for load balancing and congestion control (link selection), or duplicating the user traffic via the available communication paths for reliability (packet duplication). It is worth mentioning that it is not possible to simultaneously aggregate and duplicate data for the same DRB.

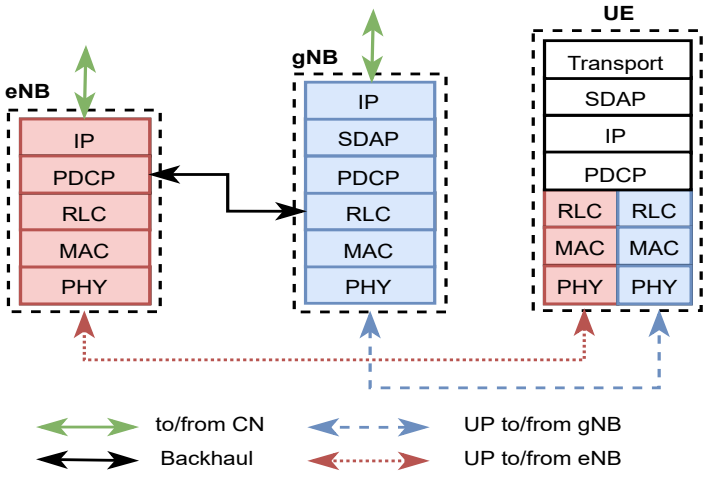


FIGURE 2.5: User plane connectivity for the EN-DC.

2.4.3 Signaling Mobility Management Aspects

During mobility within a typical MR-DC scenario, in which several small cells are deployed in the coverage area of a macro cell, as shown in Fig. 2.6, the HO events can occur in both BSs. However, they are more frequent for the SN than for the MN. Therefore, if the UE leaves the coverage area of the serving SN (S-SN), as depicted in Fig. 2.6, an SN change procedure [6] is initiated either by the MN or SN to avoid losing network connectivity via the SN's communication path. According to the 3GPP, four mobility management scenarios are possible in MR-DC operation [6]:

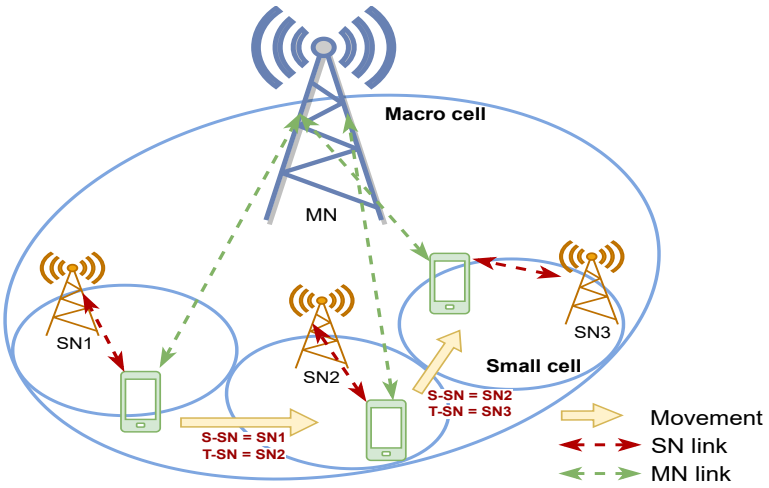


FIGURE 2.6: SN change events in a typical MR-DC deployment scenario.

- SN change (MN/SN initiated). The MN is not modified and the UE continues with the MR-DC operation.
- Inter-MN handover with/without SN change. The MN is modified, but the UE can continue with the MR-DC operation if the SN is not changed.
- MN to eNB/gNB change. The UE switches to SC operation.
- eNB/gNB to MN change. The target MN adds an SN during the HO. Hence, the UE switches from SC to MR-DC operation during this procedure.

In an SN change scenario, when the new SN, i.e., the target SN (T-SN), confirms the allocation of radio resources for the UE, the MN sends either a *SN Release Request* or *SN Change Confirm* message to the S-SN indicating that it must stop the communication with the UE [6]. Consequently, the UE temporarily communicates with a single BS, i.e., the MN, until the RA procedure with the T-SN is completed. The message exchange of the SN change procedure initiated by the MN and SN is illustrated in Figs. 2.7 and 2.8, respectively. In this regard, the UE stops communicating via the SN from step 3a to step 9 for the MN-initiated case and from step 6 to step 10 for the SN-initiated case. It is worth mentioning that it is impossible to use the DAPS HO to address the handover event in the SN since both radio interfaces are already in use.

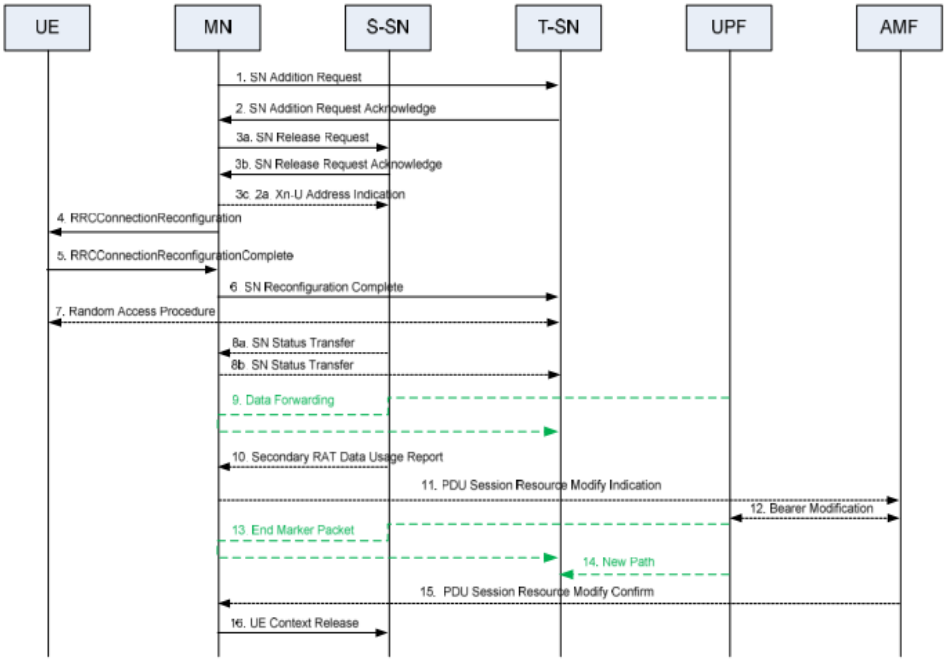


FIGURE 2.7: Signaling exchange for the SN change MN-initiated [6].

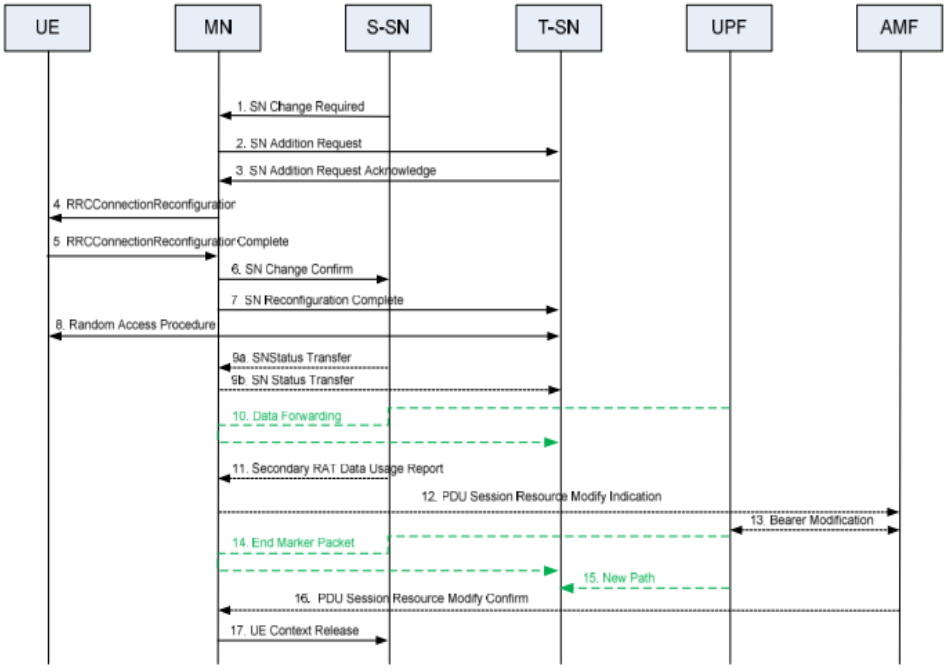


FIGURE 2.8: Signaling exchange for the SN change SN-initiated [6].

2.4.4 MR-DC Operation Challenges

The key challenges to consider for an efficient MR-DC operation lie in how the incoming PDCP PDUs are split through the MN and SN, along with the method used to reorder the PDUs before they are delivered to the upper layers. These aspects are detailed as follows.

2.4.4.1 Packet Reordering and Discarding

The communication paths associated with different BSs add distinct delays to the traffic they carry. This results from the heterogeneity of radio link conditions, the difference in the assigned radio resources, and the procedures that each BS employs to tackle the data transmission, which may depend on the used RAT. The delay difference can cause out-of-order arrival of PDUs, which may harm the performance of reliability oriented transport layer protocols such as TCP. The 3GPP specifies a PDCP-level reordering mechanism for MR-DC operation, which intends to address the out-of-order reception. For this, a reordering timer and a reordering window are used at the receiver PDCP layer to wait for any delayed PDU(s) whenever a PDU sequence number gap in the reordering window is detected [8].

Suppose the missing PDUs are not received before the reordering timer expires. In that case, the delayed PDUs will no longer belong to the reordering window provided to the upper layers. Therefore, the PDUs present in the reordering buffer are delivered to the upper layers, creating data with sequence gaps that may affect the performance of the upper layers. To mitigate this problem, the reordering timeout value configured for this timer should compensate for the delay difference between communication paths, e.g., the delay added by the backhaul. However, the 3GPP specifies neither a particular timeout value nor a procedure for choosing one. Indeed, an inadequate timeout value choice can even degrade the obtained aggregate throughput. For instance, an immense timeout value can help compensate for delays incurred during HOs or RLFs. Nonetheless, it can lead to an excessive waiting time, at the PDCP layer, that causes a bufferbloat. On the other hand, a small timeout value can move the reordering window too early, which causes spurious PDU discards.

2.4.4.2 Flow Control Logic

In MR-DC operation, the transmitting PDCP layer determines the amount of PDCP PDUs to transmit via each communication path. For this, it uses a flow control mechanism, which should dynamically split the user traffic according to the assigned radio resources and radio link conditions experienced at the MN and SN. In this regard, achieving a maximum aggregate throughput requires the flow control to maintain the transmitting RLC buffers with sufficient data to prevent under-utilized links while avoiding congestion that can increase the PDUs' sojourn times [9, 27]. Likewise, as much as possible, the flow control should minimize the PDU reordering needs at the receiver.

2.4.4.3 MN/SN Change Interruption Time

Both BSs simultaneously transfer UP traffic to the UE during data aggregation. Hence, the UE will no longer receive data via one communication path if a HO is required either in the MN or SN. For example, considering the typical delays for the CP and UP of eMBB services [28], the backhaul link, and the different steps involved in the SN change procedure, as shown in Figs. 2.7 and 2.8 and Table 2.3, the communication between the UE and SN is interrupted for approximately 74 ms for the SN-initiated case and 79 ms for the MN-initiated case. The interruption period can be longer if any of the steps of the SN change procedure cannot be completed, they take longer, or the backhaul latency is larger. In this regard, such an interruption period can cause the application to stop receiving data, challenging meeting the KPIs targets defined for the application.

TABLE 2.3: Typical delays for the SN change procedure SN-initiated, adapted from [29, 30].

Task	**Typical Delay [ms]**
(1) RRCConnectionReconfiguration	6
(2) SN Change Confirm	5
(3) UE applies new configuration	15
(4) RRCConnectionReconfigurationComplete	12
(5) SN Reconfiguration Complete	5
(6) Random Access Procedure	40
(7) Data from T-SN	1
SNC-IT = (1) + (3) + (4) + (5) + (6) + (7) - (2)	74

2.4.4.4 MN/SN Failures

In MR-DC operation, when the UE experiences an RLF on the MN, the UE triggers a Fast MCG Recovery procedure or an RRC Connection Reestablishment procedure [31]. In this case, the UE stops the communication for all radio bearers configured in the MN and sends a *MCGFailureInformation* [31] message to the MN via the SN. Depending on the information contained in the message, the MN can decide whether to change the connection with the UE to a better cell or to release it [6, 32]. On the contrary, upon a failure in the SN caused by an RLF or an SN addition/change failure, the connection between the UE and SN is suspended, and the UE notifies the MN with an *SCGFailureInformation* message [31]. Then, the MN decides whether to release or change the failing SN. Since the connection with the MN is not affected, the UE can still receive the user's data via this link [32]. However, any pending packets at the SN may be lost during an SN change.

Furthermore, for the MN failure case, the UE will not receive the user's data via the MN link for approximately 30-70 ms [33], which is comparable with a typical HO interruption time. On the other hand, when the SN fails, the data interruption time can take significantly longer since the 3GPP does not specify any fast recovery method. In this regard, during the experienced data interruption time, PDCP PDU losses or out-of-order data reception can occur. Thus, the application will stop receiving data for a period that varies depending on the reordering timeout value and transport layer protocol in use.

2.5 MC Challenge Showcase

To quantitatively illustrate the importance of efficient *Packet Reordering* and *Dynamic Flow Control* methods for the performance of the upper layer protocols, we implement and evaluate Dual Connectivity on an LTE testbed. This PDCP-MC solution has been chosen since it is the preferred method for the first commercial deployments of 5G through the EN-DC solution [7]. For our evaluations, the UP functionalities of Dual Connectivity with the split bearer option [6] are implemented using the LTE/5G-NR compliant Open Air Interface software for BS, UE, and CN [34]. In the following, a complete description of the testbed and the showcase results are presented.

2.5.1 Software Implementation of DC

We extended the existing protocol stack implementation of OAI for the BS and UE to support the UP functionalities of LTE-DC with split DRB for the downlink. For this, we implemented in the PDCP layer of the UE the 3GPP reordering mechanism described in [8]. Moreover, we incorporated new functionalities in the BS software to support the operation of distinct flow control algorithms. The code used in the BS, UE, and EPC for the experiments can be found at *https://github.com/Carlitops/DC_Flow_Control*. The architecture of the implemented testbed is depicted in Fig. 2.9 and described as follows.

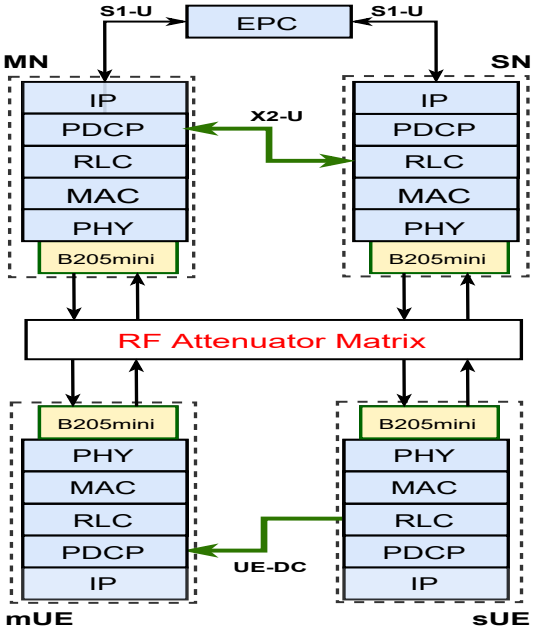


FIGURE 2.9: LTE-DC testbed architecture.

2.5.1.1 Software Implementation for BS

At the BS side, we implement the X2-U network interface to handle the transfer of PDCP PDUs from the BS acting as MN to the one serving as SN. For this, we use UDP encapsulation instead of the GPRS Tunnelling Protocol. Given the negligible protocol header overhead difference, this simplification does not affect the experimental results. Therefore, when the MN's X2-U interface receives a PDU from the transmitting PDCP layer, it sends the PDU through the UDP encapsulation to the SN's X2-U. Once the PDU arrives at the SN, the X2-U interface forwards the PDU to the RLC layer for subsequent transmission towards the UE through the Uu interface. Furthermore, we use the X2-U interface at the SN side to send flow control statistic messages to the MN. In this regard, the report message is first sent to the X2-U interface and then forwarded to the MN using the UDP encapsulation. Once a report message arrives at the MN, the X2-U delivers it to the PDCP layer for further processing. In this setup, the X2-U latency equals the backhaul latency, and the X2-U link capacity is approximately 1 Gbps.

2.5.1.2 Software Implementation for the UE

The MR-DC operation requires the UE to have one common PDCP layer for the lower layers corresponding to the two BSs, i.e., the RLC, MAC, and PHY protocol stacks, as illustrated in Fig. 2.5. These two lower stacks can operate simultaneously to aggregate traffic from the MN and SN. Since the current implementation of the OAI's UE does not support such functionality, we integrate two OAI UE instances to mimic the UE in DC operation. As it is depicted in Fig. 2.9, we refer to the UE instance connected to MN as *mUE* and the UE connected to SN as *sUE*. Both UEs are connected using the *UE-DC* interface, defined in this thesis for such a purpose. In addition, the UE-DC interface works similarly to the X2-U interface, i.e., it transports PDUs from the sUE to the mUE using a UDP encapsulation. Moreover, the delay added by the communication link between sUE and mUE hosts in our testbed is approximately 0.15 ms one-way, which has a negligible impact on our evaluations.

2.5.1.3 Testbed Setup

Five hosts from the ORBIT Testbed [35], connected using a Gigabit-Ethernet switch, are used to represent the UEs, BSs, and CN. The hosts have an Intel Core i7-4770 CPU @ 3.4GHz processor, 16 GB of RAM, and Ubuntu 16.04.1 with 4.15.0-52-low-latency kernel installed, which is one OS/kernel combination that OAI supports. Additionally, four Software-Defined Radios (SDRs), model USRP B205mini, are connected to the pair BS-UE hosts. Each SDR is electromagnetically isolated from the others, but they are connected using a programmable RF attenuator matrix, model JFW 50PMA-012 [35]. This setup allows having an isolated single input single output RF path between each BS and UE, which leads to a negligible inter-cell interference. Further general configurations and parameters used for the BSs are illustrated in Table 2.4.

TABLE 2.4: General configuration for the BSs

Parameter	Value
Duplex Mode	FDD
E-UTRA Band	7
DL Frequency for MN	2.68 GHz
DL Frequency for SN	2.63 GHz
Bandwidth for MN	5/10 MHz
Bandwidth for SN	5/10 MHz
Max number of HARQ transmissions	4
RLC mode	UM

2.5.2 Showcase Results

We study the performance of TCP and UDP protocols for downlink traffic in three scenarios: (i) single connectivity (SC), (ii) DC without packet reordering function (DC_NoR), and (iii) DC with packet reordering function, DC_Reo, using the 3GPP reordering algorithm given in [8]. Since 3GPP does not specify a concrete value for the reordering timeout, values of 40, 60, 80, 100, and 150 ms have been evaluated. Moreover, a simple flow control logic based on a Round Robin traffic distribution is used at the MN. For this, the MN, denoted as SC_A, and the SN,

denoted as SC_B, are connected using the X2-U interface, which adds a fixed delay of 10 ms. Furthermore, for realistic analysis and to demonstrate that the aggregate throughput is affected by the variance in channel capacities, we use a radio link channel trace from a pedestrian user obtained from [36]. In this regard, each BS uses different CQI sets, the values of which change every second and produce different throughput results. The throughput obtained in the SC scenario, illustrated in Figs. 2.10 and 2.11, serve as a baseline to benchmark the aggregate throughput obtained in the DC cases. For both TCP and UDP analysis, the traffic is generated using the *iperf3* tool in sessions of 30 seconds.

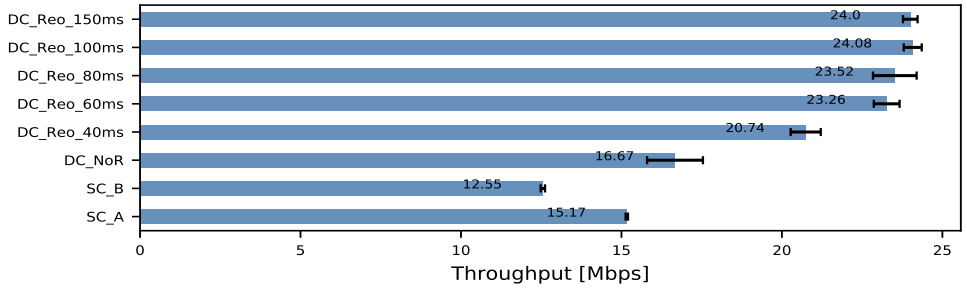


FIGURE 2.10: Throughput analysis for TCP traffic.

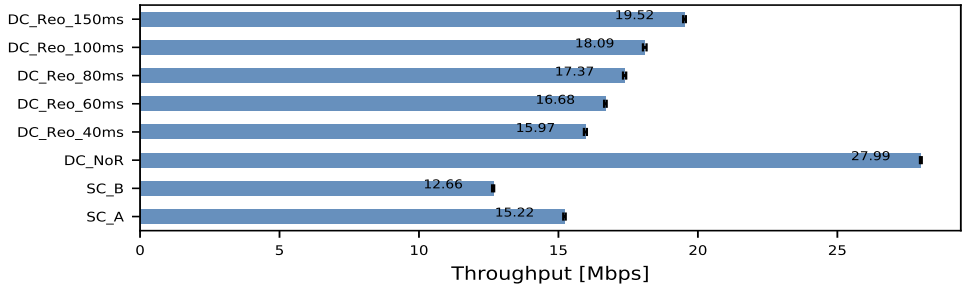


FIGURE 2.11: Throughput analysis for UDP traffic.

For TCP traffic, the results presented in Fig. 2.10 for the DC_Reo case show that the aggregate throughput is on average 36% to 58% higher than the one achieved by SC_A. However, it is lower than the ideal aggregate throughput, i.e., 27 Mb/s. Similarly, in the DC_NoR case, the aggregate throughput is only 9.8% higher than the one achieved by SC_A. In the latter, the out-of-order packets make TCP go continuously into the congestion avoidance phase. Hence, causing fast retransmissions and/or retransmissions that degrade the throughput performance. Additionally, in the DC_Reo case, the aggregate throughput depends on the chosen reordering timeout value. If this value is not enough to compensate for the delay differences between the communication paths, out-of-order packets are delivered to the upper layers. On the contrary, a large reordering timeout may cause the packets to wait excessively in the PDCP reordering buffer. This issue increases the end-to-end delay, which can trigger TCP retransmissions by timeout. In both cases, an inadequate timeout value negatively affects the throughput performance.

For UDP traffic, as illustrated in Fig. 2.11, the ideal aggregate throughput is achieved in the DC_NoR scenario, i.e., 27 Mb/s. However, when the reordering function is in use, i.e., DC_Reo case, the obtained throughput is lower than the one obtained in DC_NoR. Since UDP is not a reliability-oriented protocol, it always sends a fixed amount of data controlled by the application. This behavior, along with the flow control logic used in this experiment, results in most packets arriving at the UE through the faster communication path. Hence, placing them in the PDCP reordering buffer is necessary until the delayed ones arrive or the reordering timeout expires. Packet reordering can reduce the throughput. Nevertheless, for some UDP-based applications,

where the out-of-order packets degrades the application performance (e.g., in a video conference) a delay-limited packet reordering can be beneficial.

For both TCP and UDP, the results show that the MC aggregate throughput is significantly influenced by how the differences in terms of latency, radio link conditions, and channel capacity between communications paths are managed by the flow control and packet reordering mechanisms. Indeed, all types of MC solutions' performances are expected to be affected in varying degrees due to the differences mentioned above. DC can significantly boost the throughput for UDP traffic but at the cost of delivering out-of-order packets, which can affect the performance of applications. Similarly, TCP performance can be improved by the packet reordering process but with a benefit that depends on the reordering timeout value.

2.6 Summary

In this chapter, we have discussed the opportunities and challenges that MC offers to effectively utilize the system resources to enhance the user throughput, increase the data reliability, and reduce the adverse effects of handover. Additionally, we have presented and discussed different MC architecture options, categorized by the protocol layer where the user traffic is split or duplicated. We have also described several MC standardized solutions; specifically, we have focused on the 3GPP MR-DC technology. Albeit the system enhancements MC promises, it also defines several challenges for its efficient implementation. Following the challenges described in this chapter, we have shown through experimental evaluations of TCP and UDP traffic that the design decisions such as the flow control and packet reordering schemes can significantly impact the overall system performance.

Chapter 3

Addressing the Data Aggregation Problem in MR-DC

This chapter presents the challenges encountered by the transport and application layers to efficiently aggregate data in MR-DC operation. For this, we study the main factors that affect the data aggregation and we propose a solution to approximate the ideal aggregate throughput. First, we review the different strategies used by state-of-the-art solutions to increase the per-user data rate when the UE is simultaneously connected to two BSs. Next, targeting the gap on this topic, we present a novel flow control algorithm that allows the UE to approximate the ideal aggregate throughput. Lastly, to realistically validate the performance of the proposed algorithm, the results of different evaluations performed against benchmarking and state-of-the-art algorithms in a LTE/NR testbed are presented and discussed.

Contributions:

- C. Pupiales and I. Demirkol, "Efficient Traffic Aggregation for Dual Connectivity," in OAI Virtual Workshop 2021, Jun 2021 [14].
- C. Pupiales, D. Laselva, and I. Demirkol, "Capacity and Congestion Aware Flow Control Mechanism for Efficient Traffic Aggregation in Multi-Radio Dual Connectivity," in IEEE Access, vol. 9, pp. 114 929–114 944, August 2021 [9]. (Area: Telecommunications; Rank: 43/94; Quartile: Q2; IF: 3.476).

3.1 Problem Description

Data aggregation in MR-DC operation for downlink traffic implies splitting PDCP PDUs at the MN's PDCP layer and then reversely aggregating them at the UE's mirroring layer. Since these PDUs are independently transmitted via the MN and SN using the split DRB, i.e., a bearer configured at both BSs, the aggregate throughput would equal the sum of the throughputs achieved at the MN and SN when employing SC. In this regard, it would be beneficial to use MR-DC instead of SC if the obtained aggregate throughput is higher than the highest SC throughput achieved at each BSs regardless of the transport layer protocol and application in use.

When the UE is connected to two different BSs, the RRM procedures such as packet scheduling and link adaptation are managed at each BS according to the experienced radio link conditions and load level at each Transmission Time Interval (TTI) or slot. Consequently, the PDUs experience different sojourn times depending on each communication path's delay. Unfortunately, the sojourn time difference between communication paths causes out-of-order reception of PDCP PDUs, obligating the receiving PDCP layer to use a reordering mechanism to ensure in-sequence packet delivery to the upper layers.

The sojourn time difference between communication paths is ruled by the RLC buffering delay and the backhaul delay. The former comes from the fact that due to the variability of the radio link conditions and assigned radio resources, each BS can transmit just part of the data placed at its RLC buffer at each transmission opportunity, i.e., at each TTI. Thus, the remaining data must wait in the RLC buffer until the next transmission opportunities. On the other hand, the backhaul delay is a consequence of the physical connection between BSs, which has limited capacity, and a non-zero delay. Since the RLC buffering delay varies along the time, the sojourn time experienced by PDUs at each communication path would also change. Consequently, the percentage of received out-of-order PDCP PDUs may increase or reduce according to such variability.

At a glance, the out-of-order reception of PDUs may not represent any problem at the PDCP level. Nevertheless, the out-of-order packet reception can significantly affect TCP-based traffic performance at the transport and application level. Indeed, the aggregate throughput can even be lower than the one achieved in SC, as we showed in [13]. Unlike TCP, the UDP throughput is not affected by this problem. However, the application can treat the out-of-order deliveries as packet losses. Thus affecting the perceived quality of delay-sensitive applications using UDP [37].

The PDCP-level reordering mechanism [8] aims to minimize the out-of-order deliveries to the upper layers. Hence, such a mechanism waits for any delayed PDU at a time equal to the value configured for its reordering timer (t-Reordering) [8]. A large *t-Reordering* value can help to ensure the in-sequence delivery to the upper layers. However, since the transport layer is unaware of the delay added by the reordering mechanism, the TCP transmitter can erroneously retransmit the non-acknowledged data, i.e., it reacts with a retransmission by timeout, reducing the TCP's congestion window and thus, the aggregate throughput. On the contrary, a short *t-Reordering* value cannot be enough to compensate for the delay difference between PDUs with consecutive sequence numbers transmitted via each communication path, respectively. Thus, the receiver PDCP layer may discard the delayed PDU(s) too early.

Besides the packet reordering required at the receiver PDCP layer to avoid affecting the performance of the upper layers, the MN and SN should maintain a continuous data flow with the UE to maximize the aggregate throughput. For this, the transmitting PDCP layer should split the incoming traffic so that the BSs have sufficient data to be transmitted at each TTI, but without increasing the RLC buffering delay. Nevertheless, the MN's PDCP layer is unaware of the variability of the radio link conditions and RLC buffering delay, especially the ones from the SN. Thus, the splitting decisions become challenging. For instance, splitting the incoming PDCP PDUs based on a Round Robin logic can fairly utilize the resources in both BSs. Nonetheless, this logic may be helpful only when the radio link conditions and the instantaneous data rates are approximately the same between the UE and both BSs.

3.2 State-of-the-Art Solutions

Most of the available flow control algorithms are designed to choose either one or both communication paths with the aim of reducing the end-to-end latency, maximizing the throughput of the MN or SN, or guaranteeing to achieve at least a minimum throughput for all users in the MN and/or SN's cell. For instance, in [38], the traffic is split using a fixed ratio that does not consider the dynamism of the wireless link. The BS with the largest capacity transmits a given percentage of the incoming traffic, and the other BS, the remaining one. This fixed approach results in inefficient decisions for most of the potential scenarios. In [39], the MN sends data traffic to the SN in a request-and-forward manner. That work aims to maximize the data rate of the users connected to the SN instead of aggregating traffic from both BSs. For this purpose, the SN maintains its buffer with enough data to be scheduled at each transmission opportunity. Hence, the SN sends the data requests to the MN based on a trade-off between the buffering time and the possibility of link starvation.

Moreover, the authors in [40] propose a downlink traffic scheduling method to maximize the network throughput and keep a fair distribution among the UEs connected to the SN. The traffic splitting decision is modeled as a mixed-integer linear programming problem fed by recent CQI and buffer status information from the MN and SN. The proposed method only considers that the UEs are served either by the MN or SN at a time. Hence, traffic aggregation is not possible. Additionally, a flow control method that minimizes the end-to-end delay instead of enhancing the perceived data rate is proposed in [41]. User traffic is dynamically sent through the link that offers the lowest latency. The delay experienced in the MN and SN is characterized using deterministic network calculus theory for such a purpose. Even though this mechanism offers path diversity and some throughput improvements, the UE is not simultaneously connected to both BSs. Hence, traffic aggregation is not feasible.

In addition, authors in [42] present a utility-based algorithm that splits the user traffic in order to maximize user satisfaction. For this, utility weights based on certain QoS levels are computed. Although this work provides valuable insights, the proposed algorithm does not specify the used correlation to calculate the splitting ratio between the utility weights and the QoS metrics. Furthermore, an improved version of the solution presented by [42] is introduced in [43]. The authors propose a new control message to update the splitting ratio based on the UE's feedback. This feature is designed and tested considering the insights of the flow control proposed in [42]. Hence, it also faces the same limitations already indicated for such a work. Moreover, a flow control solution that intends to reduce the blocking state in a mmWave link is proposed in [44]. This algorithm aims to offload the user traffic from one BS to another when the mmWave link is unavailable. The results show the solution's effectiveness, yet the algorithm has not been designed to provide traffic aggregation in such deployment.

Several solutions have also been proposed for the LWA, the insights from which can be used for MR-DC. For instance, in [45], the authors offer a solution that pursues intra-cell throughput fairness for TCP-based traffic. The algorithm uses the UE's feedback to estimate the delay experienced for each PDU. Subsequently, the PDUs are transmitted via the fastest link. In this regard, results show a better performance of their proposal than Multipath TCP in some of the evaluated scenarios. However, the throughput improvement compared to the LTE throughput is only appreciable if both LTE and WLAN BSs have comparable individual throughputs. Furthermore, a per-PDU delay-based algorithm based on the UE's feedback is presented in [46]. The PDUs are sent through the path that offers the lowest transmission delay to equalize the sojourn delays in both communication paths and reduce the out-of-order issues. For this, the delays observed by the previous PDUs are continuously computed using the UE status reports

from both communication paths. The evaluations are conducted only for TCP traffic in a co-located scenario, i.e., the LTE and WLAN RATs belong to the same BS. Hence, it is impossible to know what would be the backhaul delay's impact on this algorithm's performance.

The relevant research studies presented in this section propose flow control methods that do not consider their impact on the performance of the upper layer protocols, especially for TCP. They assume that the PDCP layer receives the PDUs in sequence, and hence they are delivered in the same order to the upper layers; however, this assumption is unrealistic. Moreover, the beliefs and simplifications used by the above indicated state-of-the-art solutions in their simulators to model the real-world networks may not correctly represent the heterogeneity of the communication paths, the variability of the radio link conditions, and different network protocol configurations. Indeed, the used simulators may not include a complete implementation of the transport layer protocol stacks, limiting the effectiveness of the proposed solutions when they come to work in real-world networks.

Because of these reasons, in the following, we present a novel flow control mechanism that efficiently aggregates traffic from the MN and SN and overcomes the problems mentioned in section 3.1. The proposed flow control mechanism, named Capacity and Congestion Aware (CCW), dynamically splits the incoming user traffic considering the RLC buffering delay and MAC SDU sizes statistics from both BSs. This feature makes the CCW agnostic of the MR-DC architecture option, MAC packet scheduler design, and transport layer protocol in use. Lastly, the CCW flow control does not define any new signaling feedback from the UE for the traffic splitting decisions.

3.3 CCW Principles Design

The CCW flow control mechanism aims to aggregate traffic from the MN and SN and approximate the ideal aggregate throughput for TCP and UDP traffic. For simplification, the design of the proposed solution is explained considering the downlink direction, but it is also valid for the uplink. The CCW dynamically splits the user traffic via both communication paths according to (i) the average capacity allocated by the MAC packet scheduler to the split DRB in each communication path and (ii) the average buffering delay experienced in the corresponding RLC buffers. The operation of the CCW starts once the split DRB has been configured in both BSs. Note that the CCW operates in BSs that use the same TTI value, e.g., 1 ms. However, with few modifications, it can work with any arbitrary TTI/slot duration. The main building blocks of the CCW are shown in Fig. 3.1 and described in the following.

3.3.1 Capacity and Congestion Estimation

In both LTE and NR RATs, the UE uses an aperiodic or periodic Channel State Information (CSI) report to indicate, to the BS, the instantaneous radio link channel conditions in the form of a CQI value. This value ranges from 0 to 15 and reflects the observed downlink SINR. This CQI is subsequently mapped to a Modulation and Coding Scheme (MCS) that ensures a maximum Block Error Rate (BLER) target given the SINR conditions of the UE. Then, the MCS and the assigned Physical Resource Blocks (PRBs) by the MAC packet scheduler are used to determine the corresponding Transport Block Size (TBS), i.e., the number of bytes that can be transmitted with the given BLER target at the corresponding TTI [47]. Since the radio link conditions and assigned PRBs may be continuously changing, the TBS also changes. In

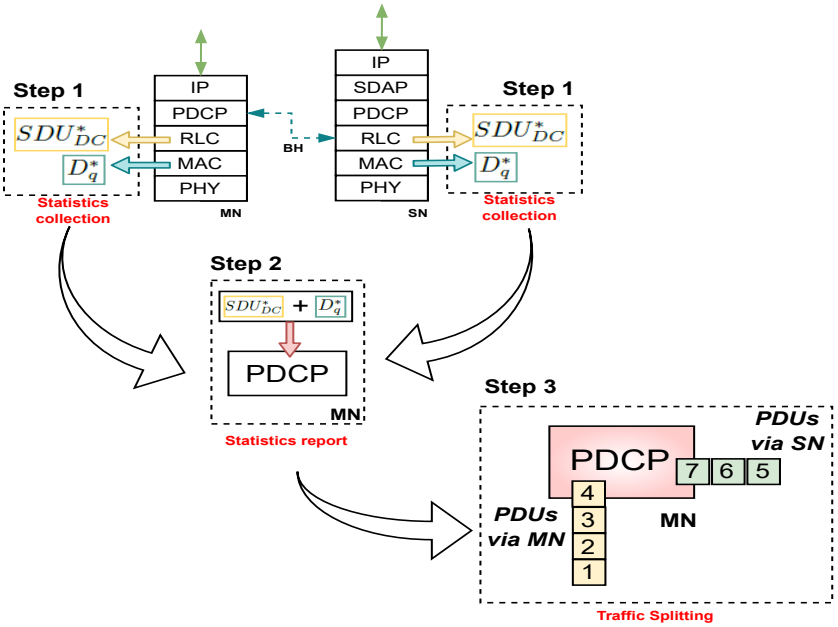


FIGURE 3.1: Main building blocks of the CCW.

consequence, fewer data can be transmitted towards the UE in poorer radio link conditions and/or higher cell loads. For this reason, unsent PDUs should wait in the corresponding RLC buffer for the following transmission opportunities, the delay of which depends on the MAC packet scheduler algorithm used in the BS [48].

Typically, the MAC packet scheduler shares the available PRBs among the active UEs and among the radio bearers configured for the UE [49], which assignment is vendor-specific. Regardless of the logic used to distribute the PRBs, the number of bytes assigned by the MAC packet scheduler to each DRB corresponds to the MAC SDU size. Indeed, it also represents the number of bytes to be pulled out for transmission from the corresponding RLC buffer at a given TTI [50]. It is worth mentioning that multiple MAC SDUs corresponding to the same or different DRBs may be part of a single transport block. Therefore, to make our CCW flow control algorithm simple and transparent to the RAT and MAC packet scheduler design, we use the split DRB's MAC SDU sizes to compute the number of PDCP PDUs to be transmitted via the MN and SN according to the following procedure.

First, we determine the effective number of bytes that can be pulled out from the RLC buffer for the split DRB at the TTI n, the value of which corresponds to the MAC SDU size and is denoted as $SDU_{DC}[n]$. Since the MAC SDU size can significantly change from one TTI to another, we use the Exponentially Weighted Moving Average (EWMA) [51] for averaging the SDU_{DC} values and hence, reduce the bias against these abrupt changes. Therefore, the EWMA of SDU_{DC} is finally used to determine the amount of PDCP PDUs that can be forwarded to an RLC buffer in each BS, and it is also calculated at each TTI n as follows

$$SDU^*_{DC}[n] = \alpha \times SDU_{DC}[n] + (1 - \alpha) \times SDU^*_{DC}[n-1], \tag{3.1}$$

where $SDU^*_{DC}[n]$ is the computed EWMA value, in bytes, of $SDU_{DC}[n]$ at the TTI n, α is the EWMA's *smoothing factor*, and $SDU^*_{DC}[n-1]$ is the previous SDU^*_{DC} value.

Additionally, the existing buffering level in the RLC buffer can be represented as the time that certain PDUs wait in the buffer before being transmitted, i.e., buffering delay [52]. Therefore, the RLC buffering delay (D_q) of the split DRB is defined at each TTI n according to

$$D_q[n] = \frac{RLC_{BS}[n]}{SDU_{DC}[n]}, \tag{3.2}$$

where $RLC_{BS}[n]$ is the RLC buffer size in bytes measured in the TTI n for the split DRB. Since the instantaneous D_q value can also change from one TTI to another, we use the EWMA for smoothening that value as follows

$$D_q^*[n] = \alpha \times D_q[n] + (1 - \alpha) \times D_q^*[n - 1], \tag{3.3}$$

where $D_q^*[n]$ is the computed EWMA value of $D_q[n]$, in milliseconds, at the TTI n, and $D_q^*[n-1]$ is the previous D_q^* value. Note that the capacity and congestion estimation starts independently at each BS when the split DRB has been configured in the corresponding BS. Fig. 3.2 illustrates the representation of SDU_{DC} and its EWMA-based averages, i.e., SDU_{DC}^*.

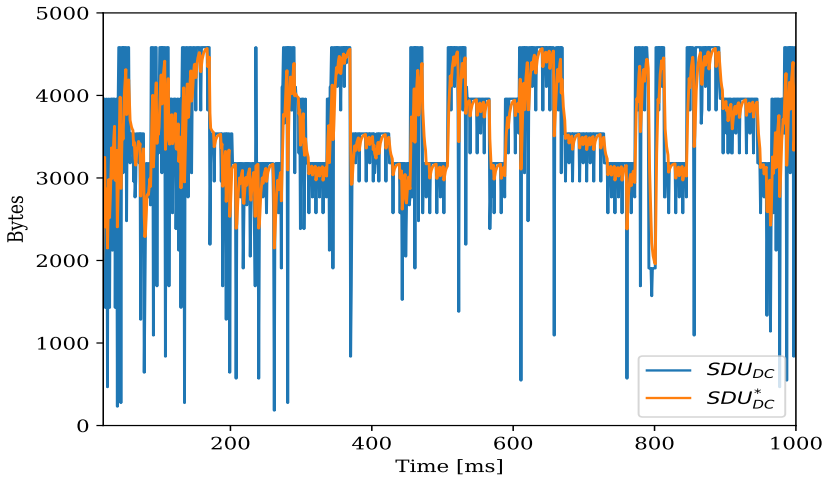


FIGURE 3.2: Estimation of SDU_{DC}^*.

3.3.2 Capacity and Congestion Report

Once the split DRB has been appropriately configured and activated at each BS, the MAC and RLC layers of each BS collect up-to-date values of SDU_{DC}^* and D_q^*, respectively. Then, these values are transmitted to the MN's PDCP layer using a capacity and congestion report (CC_R) message. Actually, for the MN, the CC_R message can be directly sent from the RLC and MAC layers to the PDCP layer. However, for the SN case, the CC_R message is transmitted through the $Xn/X2$ interface using the DDDS report, which the 3GPP has specified for flow control report purposes in [26]. Due to the two BSs being physically separated but connected through the backhaul, the CC_R message coming from the SN would have a delay equal to the backhaul latency. In this regard, it should be transmitted every TTI to avoid further delay in such a report. Nevertheless, this significantly increases the backhaul traffic and its capacity requirement. Moreover, the availability of up-to-date CQI values depends on the CSI reporting type configured for the UE, i.e., periodic or aperiodic. For this reason, the capacity and congestion report periodicity (Δt_{cc}) is dimensioned based on a trade-off between excessive

reporting signaling and freshness estimation of SDU^*_{DC} and D^*_q. The procedure described in Algorithm 1 is performed independently at each BS to send the CC_R message.

Algorithm 1 Capacity and Congestion Report Algorithm

Input: SDU^*_{DC} and D^*_q
Output: CC_R message

```
Set Δt_cc
Start TimeCounter
while TimeCounter is active do
    if TimeCounter ≥ Δt_cc then
        Collect the latest SDU*_DC and D*_q values
        Fill and send the CC_R message
        Reset TimeCounter
    else
        Keep incrementing TimeCounter
```

3.3.3 Traffic Splitting

The time-varying traffic characteristics, e.g., packet arrival rates and sizes, and the time-varying UE link conditions, e.g., the link quality and assigned resources, make it difficult for the flow control mechanism to assure continuous data flow via both communication paths. Indeed, it is challenging to cover the possible radio resources assigned to the UE while keeping the buffering delay at low levels in the corresponding RLC buffers. To tackle this challenge, the CCW defines a new buffer at the PDCP layer, where the incoming PDCP PDUs are temporarily stored in a First-In, First-Out (FIFO) manner before they are split. Therefore, it is possible to periodically split the PDUs via both communication paths according to the MAC SDU size and RLC buffering delay statistics encountered for the split DRB of the given UE. Ideally, to have the RLC buffer size of both BSs with the exact amount of data to be pulled out at TTI n, the flow control mechanism would split the UP traffic at the TTI $n-1$, and then the sent data would immediately arrive at the RLC layer. Nevertheless, the delay added by the non-ideal backhaul connection makes it difficult to achieve this for the SN. To tackle this problem, the CCW aims to have in the RLC buffers a sufficient amount of data that can be scheduled at each TTI while satisfying a buffering delay limit ($D^*_q max$) that the sent data might create in the RLC buffer. In other words, the CCW applies the principle of "keep the pipe just full, but no fuller" described in [53].

Based on the above facts, the CCW takes UP traffic splitting decisions periodically. In other words, the CCW defines the amount of data to send to each RLC layer at the beginning of every period. This period is called traffic splitting time interval (T_{CCW}) in this thesis. For this, at each T_{CCW}, the CCW estimates the amount of data that will be pulled out from the RLC buffers for transmission during a time period equal to the T_{CCW}. Note that the minimum time for the CCW to be aware of the effect of such data splitting on the SN's RLC buffering delay is equal to $2 \times BH + TTI$, where BH is the backhaul delay. Hence, having a large T_{CCW} value may be inefficient for the CCW to adapt to the radio link condition changes. Therefore, since the CC_R message arrives every Δt_{cc}, the splitting decisions should be taken every $T_{CCW} = \Delta t_{cc}$ as well. Note that we demonstrate the effect of T_{CCW} on the performance of the CCW in Section 3.5.3. Furthermore, defining and trying to satisfy a maximum RLC buffering limit, $D^*_q max$, helps to control the delay difference between the communication paths. If such delay limit is satisfied, two PDCP PDUs with consecutive sequence numbers transmitted via the MN and SN

within the same *splitting time interval*, respectively, would be received with a time difference of at most $BH + D_q^*max$.

Formally, the CCW flow control algorithm is presented in Algorithm 2, and its main tasks are detailed in the following.

Algorithm 2 CCW Splitting Algorithm

Input: $SDU_{DC}^*[MN, SN]$ and $D_q^*[MN, SN]$
Output: Number of PDUs to send through MN and SN

1: **while** $SDU_{DC}^*[MN, SN]$ and $D_q^*[MN, SN]$ are not available **do**
2: Split the PDUs using a Round Robin logic
3: Set T_{CCW}
4: Set D_q^*max
5: Place the PDUs in B_{FC}
6: **for** $b = MN, SN$ **do**
7: **if** $D_q^*max - D_q^*[b] \leq 0$ **then**
8: $ToSend_b = 0$ ▷ RLC buffer is congested
9: **else if** $D_q^*max - D_q^*[b] \geq T_{CCW}$ **then**
10: $ToSend_b = T_{CCW} \times SDU_{DC}^*[b]$ ▷ in bytes
11: **else**
12: $ToSend_b = (D_q^*max - D_q^*[b]) \times SDU_{DC}^*[b]$ ▷ in bytes
13: **if** $D_q^*[MN] \leq D_q^*[SN]$ **then**
14: $b = MN, SN$
15: **else**
16: $b = SN, MN$
17: **for** b **do**
18: $Sent_b = 0$
19: **while** $B_{FC} \neq 0$ and $Sent_b \leq ToSend_b$ **do**
20: Pull out a PDU from B_{FC} and transmit it
21: $Sent_b = Sent_b + PDU_{size}$ ▷ in bytes

- Until the first CC_R messages from both BSs are received, every incoming PDU is split using a Round Robin logic (Lines 1-2).
- Once the initial SDU_{DC}^* and D_q^* values from both BSs are available, the CCW's traffic splitter is configured accordingly. For this, the splitting time interval T_{CCW} and maximum buffering delay D_q^*max are set up (Lines 3-4).
- After the traffic splitter is configured, the arriving PDCP PDUs are placed in the FIFO buffer B_{FC} for periodic splits (Line 5).
- Every T_{CCW}, the amount of data to be sent to the RLC layer of each BS, denoted as $ToSend_b$, in computed using the corresponding SDU_{DC}^* and D_q^* values (Lines 6-12).
- Once $ToSend_b$ for both BSs is known, PDUs are pulled out from B_{FC} and sent to the corresponding RLC buffer until the total amount of sent PDUs in bytes, denoted as $Sent_b$, is greater than or equal to $ToSend_b$. Note that the communication path with the lower D_q^* is always scheduled first (Lines 13-21).

As can be observed, none of the employed parameters in the CCW flow control algorithm depend on the MR-DC architecture option, MAC packet scheduler, and transport layer protocol in use.

In addition, the statistical information required by the CCW regarding the split DRB is available at the 4G/5G BSs since a BS can handle multiple UEs configured with multiple DRBs. Indeed, each DRB has its PDCP and RLC instances [8, 54].

3.3.4 Reordering Timeout Dimensioning

As stated in Section 3.3.3, the CCW limits the RLC buffering delay in both BSs, which contributes to controlling the delay difference experienced by two PDCP PDUs with consecutive sequence numbers that were split in the same splitting interval. Hence, generally speaking, the *t-Reordering* value should be slightly higher than $BH + D_q^*max$ milliseconds. However, the *t-Reordering* value should also consider the variable sojourn delay experienced by a PDCP PDU in the LTE and NR communication paths. According to [28], the UP latency, i.e., the PDU sojourn delay, is measured as the elapsed time an IP packet would experience from entering until leaving the transmitting and receiving PDCP layers, respectively. In this regard, the ideal one-way UP latency target, i.e., without considering the RLC buffering delay and retransmissions, defined for eMBB services in NR is 4 ms [28, 47], which matches with the target defined for LTE (Rel. 10) [55].

Regardless of the MR-DC architecture option, one communication path is faster than the other. For instance, assuming an EN-DC deployment with $BH = 10$ ms, eMBB traffic, and no buffering delay, the UP latency for the MN and SN would be 4 ms and 14 ms, respectively. Therefore, if two PDUs with consecutive sequence numbers are split via the MN and SN in the same splitting interval, the reordering mechanism should wait for the delayed PDU at a time equal to the difference between both links. For this example, $t\text{-}Reordering = 4$ ms. Nevertheless, the variable radio link conditions and availability of radio resources make the UP latency vary over time. This variability can make the fastest link becomes the slowest and vice versa. Hence, the UP latency depends on the HARQ retransmissions delay, RLC thresholds, i.e., reordering and reassembly timers [54, 56], backhaul delay, and RLC buffering delay, i.e., D_q.

Considering the aspects mentioned above, the *t-Reordering* can be calculated as the time difference between the fastest and slowest links. Knowing such a delay difference beforehand is challenging since the real UP latency varies over time. Indeed, it may require that the receiver PDCP layer continuously measures the sojourn delay experienced in each communication path. Nevertheless, the *t-Reordering* can be calculated using the following alternative approach.

$$t\text{-}Reordering = S_{link} - F_{link} + D_q^*max, \tag{3.4}$$

where S_{link} represents the maximum theoretical sojourn delay, i.e., without HARQ/ARQ retransmissions, that a PDU would experience in either communication path, F_{link} is the minimum theoretical sojourn delay in either communication path. This approach considers that two consecutive PDUs are split through the MN and SN within the same splitting interval. Therefore, the *t-Reordering* value is dimensioned to work even in the worst-case scenario. The calculation follows these steps:

1. Determine the theoretical one-way UP latency ($UP_{latency}$) for both communication paths, e.g., 4 ms for eMBB traffic.
2. Identify the value of the RLC's timer in use (RLC_{timer}), i.e., reassembly or reordering timer for NR and LTE, respectively.
3. Identify the backhaul delay (BH).

4. Compute the minimum sojourn delay experienced in the MN's communication path:

$$Min_{MN} = UP_{latency}. \tag{3.5}$$

5. Compute the minimum sojourn delay experienced in the SN's communication path:

$$Min_{SN} = UP_{latency} + BH. \tag{3.6}$$

6. Compute the maximum sojourn delay experienced in the MN's communication path:

$$Max_{MN} = max(UP_{latency}, RLC_{timer}). \tag{3.7}$$

7. Compute the maximum sojourn delay experienced in the SN's communication path:

$$Max_{SN} = max(UP_{latency} + BH, RLC_{timer}). \tag{3.8}$$

8. Identify the slowest and fastest communication paths. Note that the slowest communication path is the one that has the highest sojourn delay. Likewise, the fastest is the one that has the shortest sojourn delay.
9. Set D_q^*max.
10. Compute the *t-Reordering* using 3.4.

Since dual connectivity has been designed to aggregate eMBB traffic [4, 7], we assume the following configuration to exemplify our calculations. $TTI = 1$ ms in both BSs, $BH = 10$ ms, $D_q^*max = 20$ ms, and the default values specified by the 3GPP for the RLC reordering and reassembly timers, i.e., 35 ms in both cases [54, 56]. For this, the sojourn delays in the slowest and fastest links are 35 ms and 4 ms, respectively. Hence, the *t-Reordering* would be 51 ms. Since the 3GPP specifies a set of values to use for the *t-Reordering*, the value obtained in step 10 must be approximated to the closest higher value indicated in [31]. For this example, the final *t-Reordering* value is 60 ms. It is worth mentioning that using a slightly higher *t-Reordering* value has a negligible impact on the results. However, significantly larger values can unnecessarily increase the PDCP reordering delay, thus possibly affecting the performance of the application, especially the TCP-based ones.

3.4 Evaluation Framework

To validate the proposed CCW flow control algorithm for MR-DC, we experimentally evaluate it using the LTE/NR compliant testbed developed in section 2.5.1. In addition, the experimentation is conducted using the ORBIT Testbed facilities [35]. The details of the evaluation scenarios and the obtained results are exposed in this section.

3.4.1 Benchmarked Flow Control Algorithms

To compare the performance of the CCW, we have selected and implemented two flow control solutions for a benchmark. In the first solution, incoming PDUs are split via the MN and SN according to a Round Robin (RR) approach. We use the PDCP sequence numbers for the

splitting decisions, i.e., the PDUs with even sequence numbers are transferred via the MN and those with odd sequence numbers via the SN. This simplistic approach does not require any specific configuration and provides a simple but effective benchmark for the other solutions. In the second solution, each PDU is sent through the link that offers the shortest packet delay as proposed in [46]. This *Delay-based* algorithm relies on the UE status reports from both communication paths, i.e., it employs the UE's feedback when the RLC layer is configured in the acknowledged mode. Since the current implementation of OAI only supports the RLC configured in unacknowledged mode (UM), there is no RLC level feedback from the UE. However, the UE still reports HARQ/L1 level feedback. Therefore, we use the RLC buffering delay D_q, described in (3.2), to represent the delay used in [46]. This delay is measured and reported every 5 ms to the MN's PDCP layer, the periodicity of which matches the *time between UE status reports* (Δt) used in [46]. Note that there may be details and parameters to tune for the Delay-based flow control approach that cannot be determined from [46]. Nevertheless, the design of the delay-based flow control approach is primarily implemented and configured according to the methodology and values described in [46], which are listed as follows: $\Delta t = 5$ ms, maximum queuing delay $d_{max} = 30$ ms, and fairness $\beta = 0$.

Furthermore, the *CCW* algorithm is configured using the following parameters. $T_{CCW} = 5$ms, $D_q^* max = 20$ ms, and $\alpha = 0.3$. Note that the smoothing factor, α, is heuristically configured according to [51, 57].

3.4.2 Performance Metrics

We use the *aggregate throughput* obtained in the UE as the primary metric to evaluate the performance of our proposal. For this purpose, we compare the average downlink throughputs obtained for each flow control algorithm after a data session of 30 seconds. Moreover, the aggregation benefit function (A_{ben}) [58] is used as a metric to determine how efficient are the flow control algorithms in aggregating traffic. The A_{ben} uses the obtained aggregate throughput (T_{DC}), and the throughputs obtained in the UE using SC operation in both the MN (T_{MN}) and the SN (T_{SN}) for the computation of such an efficiency. Note that these values represent the average throughputs obtained over different runs in our setup.

The definition of A_{ben} for the MR-DC case, according to [58], is detailed in (3.9)

$$A_{ben}(FC) = \begin{cases} \dfrac{T_{DC} - T_{SC}^{max}}{T_{SC}^{min}} & \text{if } T_{DC} \geq T_{SC}^{max} \\ \dfrac{T_{DC} - T_{SC}^{max}}{T_{SC}^{max}} & \text{if } T_{DC} < T_{SC}^{max}, \end{cases} \tag{3.9}$$

where FC is the flow control algorithm, $T_{SC}^{max} = \max(T_{MN}, T_{SN})$, and $T_{SC}^{min} = \min(T_{MN}, T_{SN})$. In this sense, A_{ben} illustrates how efficient a flow control algorithm is to increase the user data rate by aggregating traffic in MR-DC operation in comparison with SC. The aggregation benefit is shown on a scale from -1 to 1, where a value of 1 represents the ideal aggregate throughput and the negative values indicate that the aggregate throughput is lower than the maximum SC throughput, i.e., $\max(T_{MN}, T_{SN})$.

Secondly, we evaluate the average *RLC sojourn time* to compare the delay that the flow control algorithms create at the RLC buffers. For the *Delay-based* and *CCW* algorithms, the delay added in the flow control buffer B_{FC}, i.e., PDCP sojourn time, is also assessed. This metric

shows how the data splitting decisions affect the delay difference between communication paths and the obtained aggregate throughput.

3.4.3 Evaluation Scenarios

For the performance evaluations, we define two main scenarios. The first scenario, *scenario A*, considers that both BSs have the same channel bandwidth for the UE, i.e., 10 MHz. In the second scenario, *scenario B*, BSs have different channel bandwidths, where 5 MHz and 10 MHz are used for the MN and SN, respectively. Additionally, to evaluate the performance of the CCW as realistically as possible and to assess the adaptation of the CCW to the variance of the radio link conditions and assigned resources, we use a CQI trace collected with a drive test tool for a pedestrian mobility pattern provided by [36]. This trace includes the CQI information with 1-second granularity, i.e., the CQI value remains constant during 1 second. The CQI trace is divided into two, to be fed to the two UE instances, i.e., to mUE and sUE. Note that in MR-DC operation, each BS manages its RRM procedures. Hence, each OAI BS independently requests the OAI UE to send the CQI value using an aperiodic CSI report, which on average is performed every 20 ms. Each time a CQI is requested, the OAI UE sends the next value from the trace. Since the OAI BSs are executed independently, the CQI values used for MN and SN might vary between the runs for a given time instance in the experiments. However, this provides distinct CQI combinations for different runs, which is helpful to evaluate the flow control algorithms with different CQI data sequence combinations. Fig. 3.3 illustrates the CQI values reported to the MN and SN in one experiment.

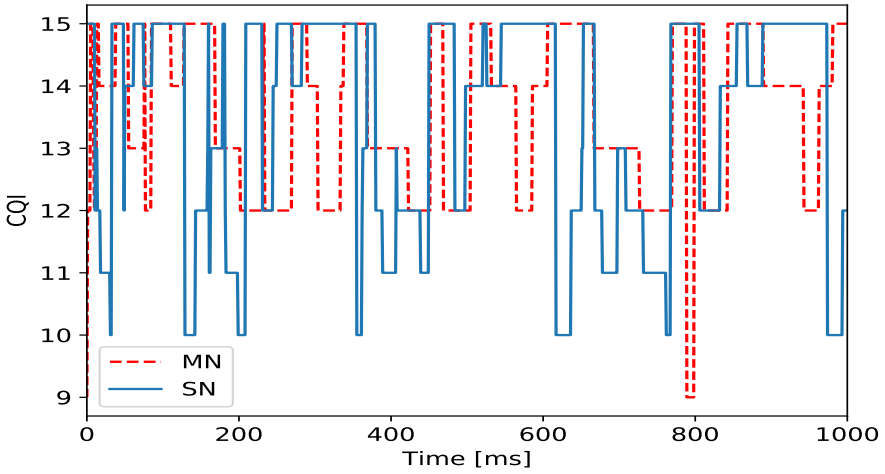


FIGURE 3.3: CQI values used at the MN and SN in an experiment.

For the evaluations, TCP and UDP traffic is generated in the downlink direction using the *iperf3* tool, which allows measuring the maximum achievable data rate. For this, the *iperf3 server* runs at the mUE's host and the *iperf3 client* at the EPC's host. In addition, the *iperf3* uses the TCP/UDP protocol stack implementations of the host operating system to transmit the generated traffic continuously. Note that the TCP's ACKs are sent to the *iperf3 client* via the MN since the uplink only uses the mUE-MN connection in our setup.

Furthermore, the impact of the PDU reordering on the obtained aggregate throughput is analyzed by enabling and disabling (NoR) the PDCP-level reordering mechanism at the UE. For the former, different values for the *t-Reordering* [8] are evaluated as shown in Table 3.1. Additionally, the backhaul latency, BH, is configured using the Linux traffic control function called

NetEm [59]. For this, in the MN's host interface, a fixed delay of 10 ms [42], [60] is added to all incoming and outcoming packets. It is worth mentioning that the same configuration is also used when the PDCP reordering mechanism is disabled. In this regard, 20 evaluations of 30 seconds duration each are performed for every analyzed configuration, i.e., per *t-Reordering* value, traffic type, and system bandwidths. Table 3.1 summarizes the different scenario setups evaluated.

TABLE 3.1: Summary of the evaluated scenarios.

Scenario	**A**	**B**
System Bandwidth (BW)	$BW_{MN} = 10$ MHz $BW_{SN} = 10$ MHz	$BW_{MN} = 5$ MHz $BW_{SN} = 10$ MHz
t-Reordering	NoR, 40, 60, 80, 100, 120, 150 ms	NoR, 40, 60, 80, 100, 120, 150 ms
PDCP Reordering window	2048	2048
Traffic Type	TCP, UDP	TCP, UDP
One-way BH delay	10 ms	10 ms

3.5 Results and Discussion

Based on the evaluations performed using the framework introduced in Section 3.4, this section presents and discusses the results in detail. To establish a baseline to compare and assess the performance of each flow control algorithm, we first evaluate the performance of TCP and UDP traffic using SC operation at each BS. For this, we use the system bandwidth configuration described for scenarios A and B. Table 3.2 illustrates the obtained SC throughputs at both BSs and the ideal aggregate throughputs for scenarios A and B, both of which serve as a baseline for further comparison.

The OAI specifications state that the maximum downlink throughput for UDP traffic in SC operation and highest CQI value is 16-17 Mbps and 34-35 Mbps with 5 MHz and 10 MHz of bandwidth, respectively. [61]. Since we use a real CQI trace, in which the CQI values change on average every 20 ms, the throughput obtained in our testbed experiments and indicated in Table 3.2 is different from the specified by the OAI. This shows the effect of such CQI variation on the obtained throughputs.

TABLE 3.2: Throughput obtained for scenarios A and B in SC and DC operation.

Traffic Type	**TCP**				**UDP**			
Scenario	A		B		A		B	
BS Type	MN	SN	MN	SN	MN	SN	MN	SN
Average SC Throughput (Mbps)	28.5	27.5	14.1	27.5	29.9	30	14.6	30
Confidence Interval (95%)	0.26	0.89	0.06	0.89	0.03	0.01	0.02	0.01
Average Ideal Aggregate Throughput (Mbps)	56 ±1.15		41.6 ±0.95		59.9 ±0.04		44.6 ±0.03	

3.5.1 Aggregate Throughput

For Scenarios A and B, we compare the performance of the CCW against the RR and Delay-based algorithms when the PDCP reordering mechanism is disabled and enabled at the UE. The results depicted in Figs. 3.4, 3.5, and 3.6 represent the average throughputs obtained for 20 distinct experiments, each with 30 seconds duration. To illustrate the variability and distribution of the obtained average throughput results, we represent our findings using a boxplot graph, in which the median can be interpreted as the average throughput obtained after the 20 experiments [62]. The visualization of the data using quartiles, where the boxplot lines represent the 25-, 50-, and 75- percentile of the obtained throughput values in 20 experiments, allows us to easily identify and compare the dispersion of the throughput obtained with each flow control algorithm and a reordering timeout value. It is important to remark that the throughput values that lie out of the boxplot's quartiles, i.e., the outliers, may be the result of individual experiments in which the different time instances between the MN and SN produce such aggregate throughput. This effect may result in more evidence for TCP traffic because the network conditions determine the reaction of the congestion control mechanisms.

3.5.1.1 Reordering Mechanism Disabled

Figs. 3.4a and 3.4b illustrate the average aggregate throughputs obtained for scenarios A and B, respectively, when the reordering mechanism is disabled.

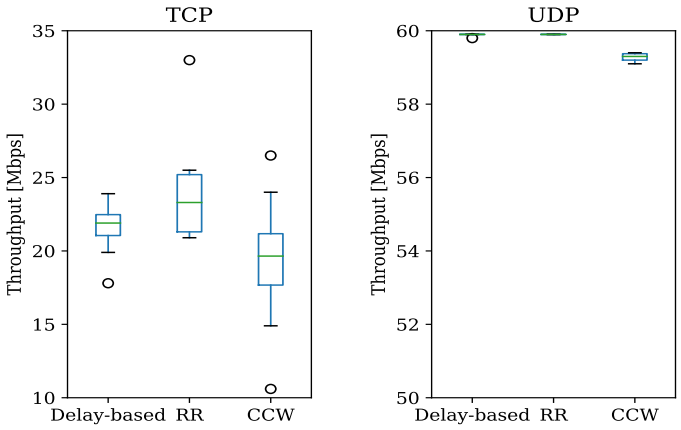
For TCP traffic, the out-of-order arrival of PDCP PDUs degrades the throughput performance in both scenarios. None of the three evaluated algorithms can overcome this issue. Indeed, they achieve less than 40% of the ideal aggregate throughput in both scenarios being assessed. Because TCP receives many out-of-order packets, it continuously reacts with *fast retransmissions*, reducing the TCP congestion window size and degrading the throughput performance.

Moreover, for UDP traffic, the three algorithms offer comparable results for Scenario A, where they achieve 59 Mbps on average. Nevertheless, approximately 49%, 48%, and 46% of the total PDUs are received out of order with the RR, Delay-based, and CCW algorithms, respectively. In addition, for Scenario B, the Delay-based and CCW are not affected by the channel bandwidth difference; thus, they achieve 44 Mbps approximately. Nonetheless, 43% and 41% of the total PDUs are received out of order, respectively. On the other hand, RR offers a lower performance because PDUs are not split according to the assigned resources at each BS. In fact, RR only achieves 37 Mbps because its inefficient splitting logic causes that 17% of the PDUs to be lost.

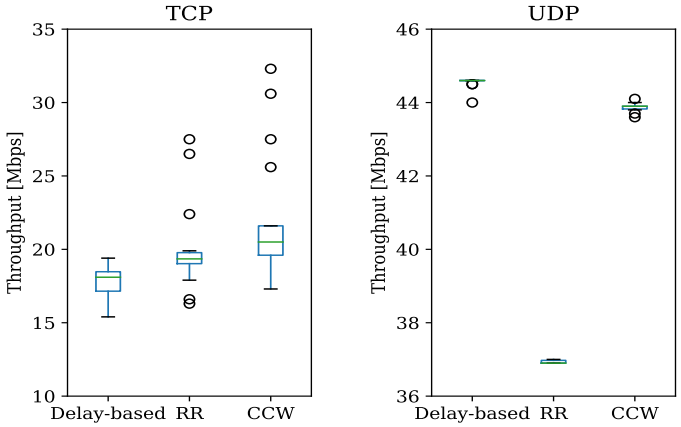
These results show that if the reordering mechanism is not used, the CCW and Delay-based algorithms are good options for aggregating UDP traffic in MR-DC.

3.5.1.2 Reordering Mechanism Enabled

The reordering mechanism is expected to help the flow control algorithms to achieve the ideal aggregate throughput for TCP traffic. Nevertheless, the CCW is the only algorithm that can achieve it regardless of the reordering timeout choice and scenario, as shown in Fig. 3.5. The RR works well only in Scenario A, as seen in Fig. 3.5a. However, in Scenario B, the RR only achieves the maximum SC throughput given by the SN, i.e., 27.5 Mbps, as seen in Fig. 3.5b. These results reflect the inability of RR to dynamically split the user traffic according to the



(A) TCP and UDP throughputs for Scenario A



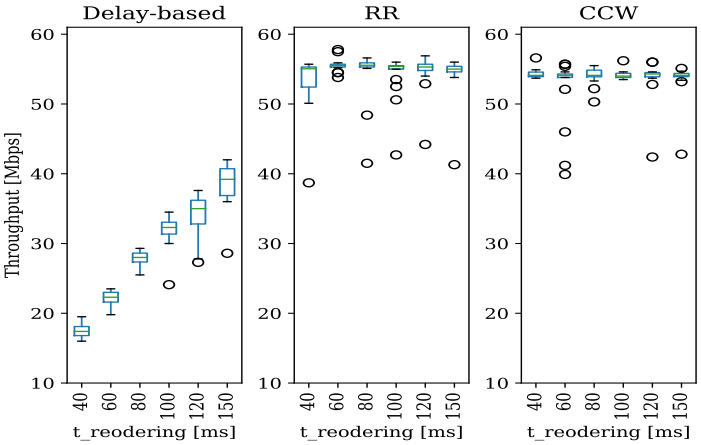
(B) TCP and UDP throughputs for Scenario B

FIGURE 3.4: Aggregate throughput obtained when the PDCP reordering mechanism is disabled.

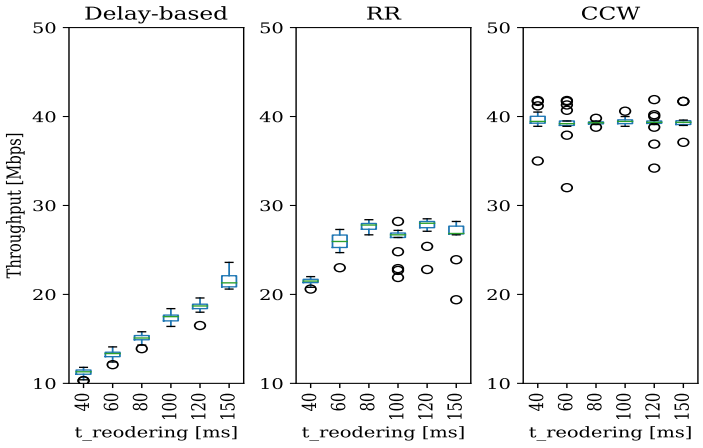
assigned resources at each BS. This simplistic method would work well only if both BSs provide similar SC throughputs, which is not always possible.

Moreover, the Delay-based algorithm bases its splitting decisions on the experienced path delay, reflecting the impact of the past splitting decisions. For this reason, the MN splits the incoming traffic via a single BS until the effect of past splitting decisions is known. This approach can make that when TCP busts appear, the delay difference between communication paths rapidly increases, the delay of which may be higher than the configured *t-Reordering* value. When this occurs, packets with discontinuous sequence numbers are delivered to the upper layers. For this reason, the obtained aggregate throughput with the Delay-based algorithm, depicted in Fig. 3.5a and 3.5b, increases with the increment in *t-Reordering* value.

For UDP traffic, Fig. 3.6a shows that the three algorithms obtain, for Scenario A, a similar throughput regardless of the *t-Reordering* choice. In this case, the continuous traffic pattern of UDP along with the same system bandwidth in both BSs facilitate the traffic splitting for the RR and Delay-based algorithms. For the latter, no bursts can affect the traffic splitting decisions. Additionally, Fig. 3.6b depicts that for Scenario B, the CCW achieves the highest aggregate throughput, i.e., 43 Mbps approximately, which, indeed, is very close to the ideal value expected for this scenario. However, the RR and Delay-based do not achieve the same performance as in Scenario A. For instance, the RR can not take advantage of the higher SN's bandwidth and transmit more PDCP PDUs through this communication path. Indeed, its simplistic splitting



(A) Average aggregate throughput for Scenario A

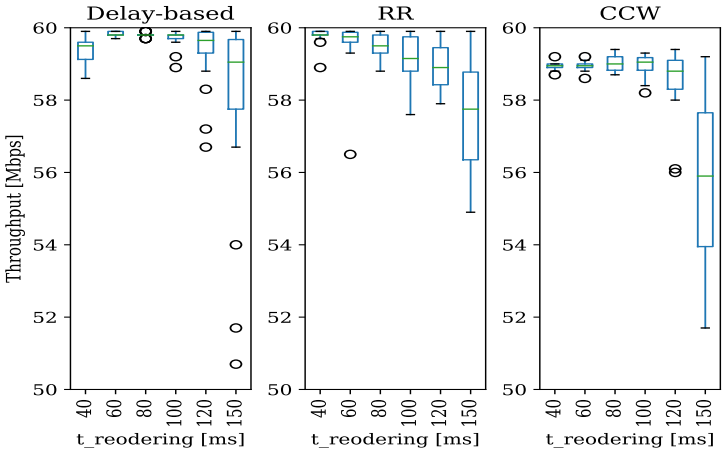


(B) Average aggregate throughput for Scenario B

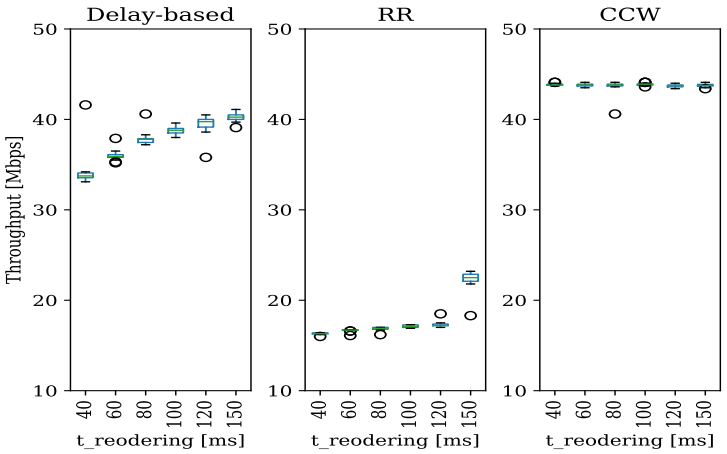
FIGURE 3.5: Aggregate throughput obtained for TCP when the PDCP reordering is enabled for different reordering timeout values.

decision logic leads to a significant increment in the delay difference between the communication paths. Therefore, more than 55% of the total PDUs arrive after the reordering timer expires.

Moreover, the Delay-based algorithm temporally buffer the PDCP PDUs if the delays of both communication paths are more significant than a maximum given value, i.e., d_{max}. When the delay in one of the communication paths is under such a threshold, all the buffered PDUs are transmitted only via one BS. This phenomenon increases the delay difference between the communication paths and causes more PDUs to arrive at the UE after the *t-Reordering* expires. Therefore, the PDUs received at the UE above this time limit are considered lost by the reordering mechanism, implying that they are not transmitted to the upper layers. Because of this, the aggregate throughput slightly increases with a higher reordering timeout value for the RR and Delay-based algorithms. Note that the d_{max} value taken from [46] and adopted in this study for the Delay-based may not be optimal in the evaluated scenarios. As shown in Figs. 3.5 and 3.6, the CCW is the only algorithm able to achieve the highest aggregate throughput for TCP and UDP traffic regardless of the reordering timeout choice and channel bandwidth combination. This critical characteristic makes our algorithm robust enough to adapt to the different conditions expected in a mobile network.



(A) Average aggregate throughput for Scenario A

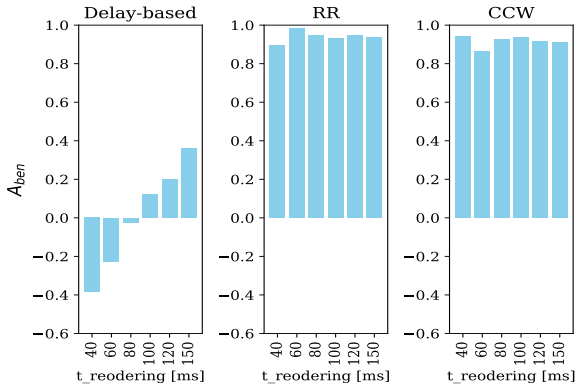


(B) Average aggregate throughput for Scenario B

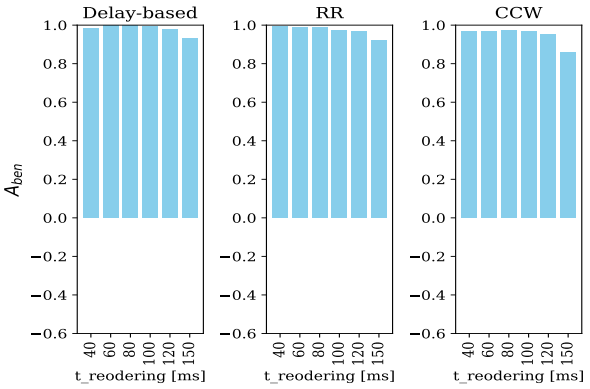
FIGURE 3.6: Aggregate throughput obtained for UDP traffic when the PDCP reordering is enabled for different reordering timeout values.

3.5.1.3 Aggregation Benefit

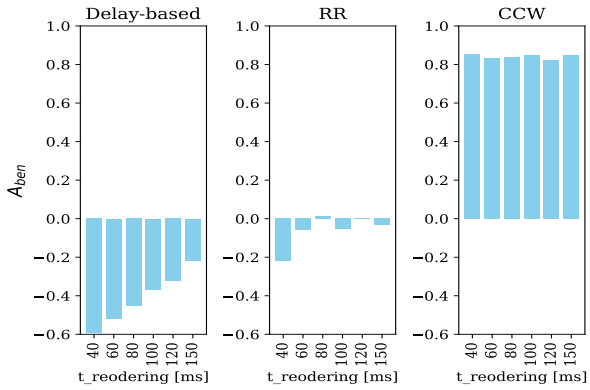
To better illustrate the CCW's robustness, we analyze the aggregation benefit A_{ben} obtained with each flow control algorithm, thus, quantifying and comparing the efficiency of each flow control algorithm to aggregate traffic in MR-DC. In this regard, Fig. 3.7 depicts the aggregation benefit obtained for the benchmarked algorithms for TCP and UDP traffic in both scenarios. The CCW flow control algorithm is the only algorithm to achieve at least 85% and 95% of the ideal aggregate throughput for TCP and UDP traffic, respectively, independently of the evaluated timeout choice and scenario. These results demonstrate the effectiveness of the scheme used by the CCW in contrast to the one used by the other two algorithms. The RR algorithm provides a comparable performance only in Scenario A, as depicted in Figs. 3.7a and 3.7b. However, as illustrated in Figs. 3.7c and 3.7d for Scenario B, the RR is extremely inefficient. Similarly, the Delay-based algorithm shows good performance only with UDP traffic in Scenario A. However, the performance is much lower for Scenario B. In addition, the aggregate throughput obtained for TCP traffic is even lower than that obtained in SC operation.



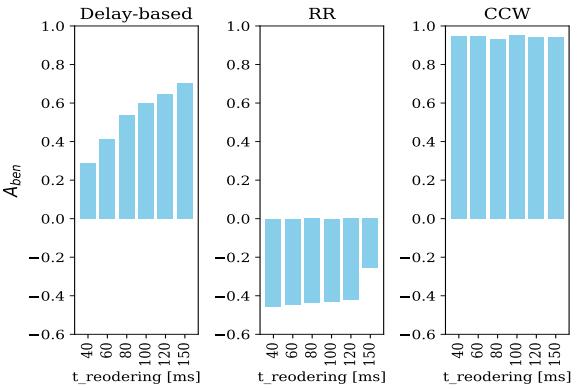
(A) A_{ben} for TCP in Scenario A



(B) A_{ben} for UDP in Scenario A



(C) A_{ben} for TCP in Scenario B



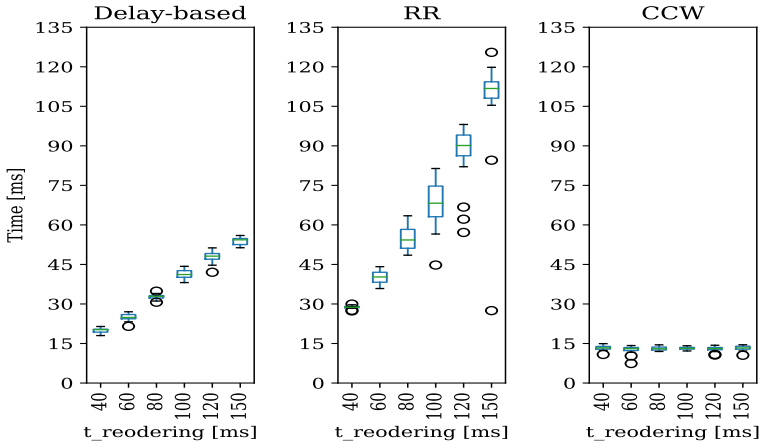
(D) A_{ben} for UDP in Scenario B

FIGURE 3.7: Aggregation Benefit comparison for Delay-based, RR, and CCW.

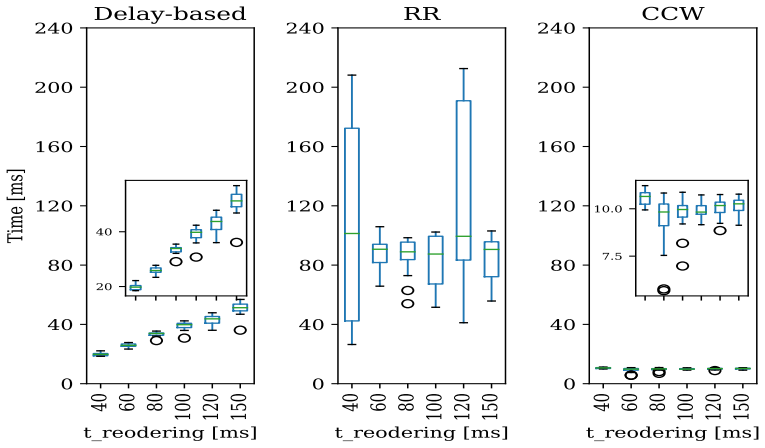
3.5.2 Sojourn Time

To study the sojourn time, we focus on the case where the reordering mechanism is enabled in the UE for both scenarios and types of traffic. This analysis is important because the delay difference between the communication paths might cause PDUs to arrive at the PDCP layer after the reordering timeout expires. Hence, these PDUs are considered lost by the reordering mechanism and are not transmitted to the upper layers. This problem creates out-of-order deliveries, which affects TCP performance, as we previously mentioned and observed in Figs. 3.4a and 3.4b. Likewise, these discarded PDUs can reduce the obtained throughput for UDP-based traffic and affect the perceived quality of such applications. In this regard, for the evaluated scenarios, we illustrate the sojourn time that PDUs experience in the RLC buffers of both BSs. In addition, the sojourn time experienced in the PDCP flow control buffer is shown for the Delay-based and CCW algorithms.

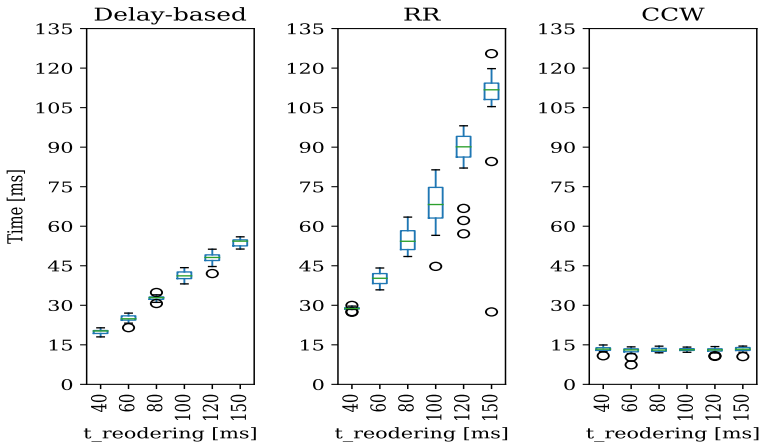
For TCP traffic, Fig. 3.8 shows that the CCW effectively limits the RLC buffering delay in both scenarios. Therefore, it is possible to maintain a continuous data flow toward the UE without significantly increasing the time difference between communication paths. Even though the use of the CCW's flow control buffer, i.e., B_{FC}, increases the packet sojourn time, as depicted in Figs. 3.9a and 3.9b, it helps to maintain an upper bound for the delay difference between communication paths. In this sense, the round-trip-time fluctuations that can arise because of the TCP bursts are minimized, especially for Scenario B.



(A) MN's RLC delay in Scenario A



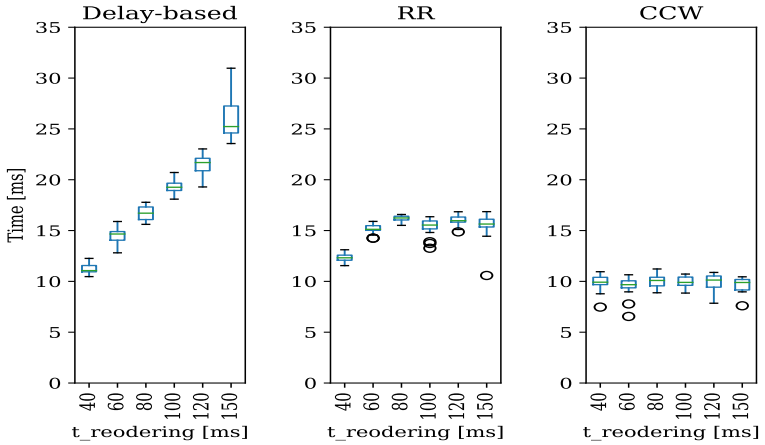
(B) SN's RLC delay in Scenario A



(C) MN's RLC delay in Scenario B

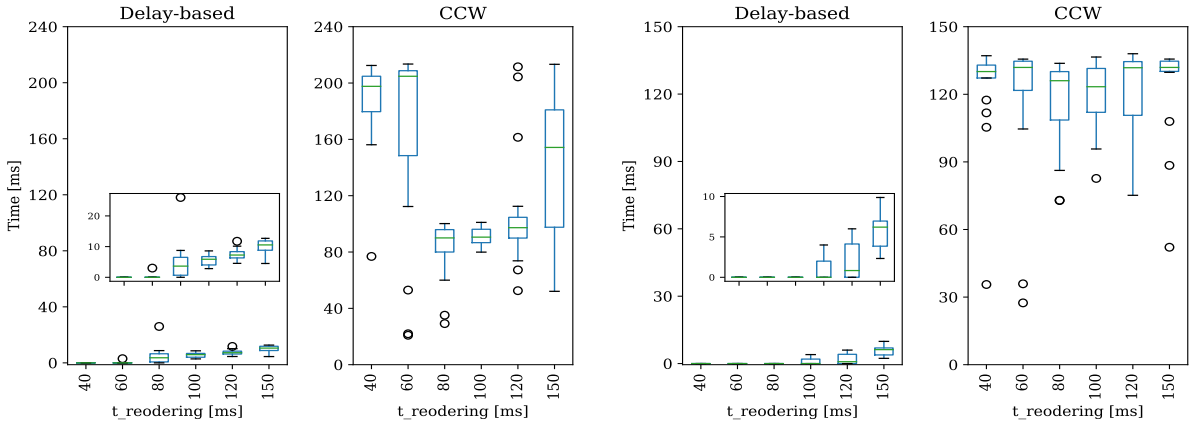
FIGURE 3.8: Average RLC sojourn times for TCP traffic using different reordering timeout values.

For UDP traffic, the CCW also maintains the buffering delay in both RLC buffers at low levels compared with the delay obtained with the other two algorithms, as shown in Fig. 3.10.



(D) SN's RLC delay in Scenario B

FIGURE 3.8: Average RLC sojourn times for TCP traffic using different reordering timeout values (cont.)



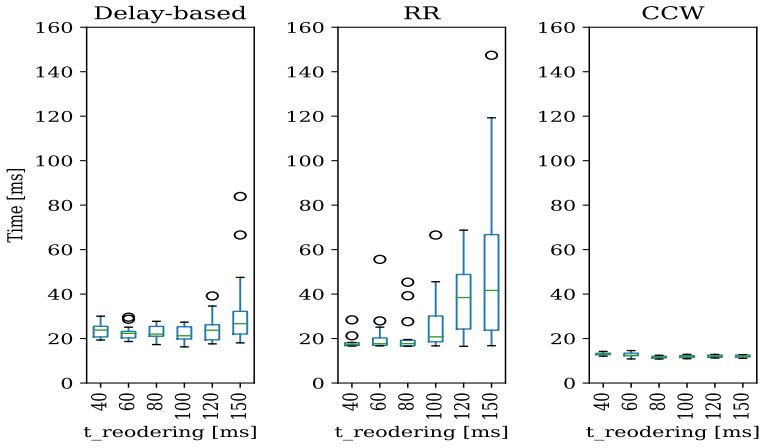
(A) PDCP delay in Scenario A

(B) PDCP delay in Scenario B

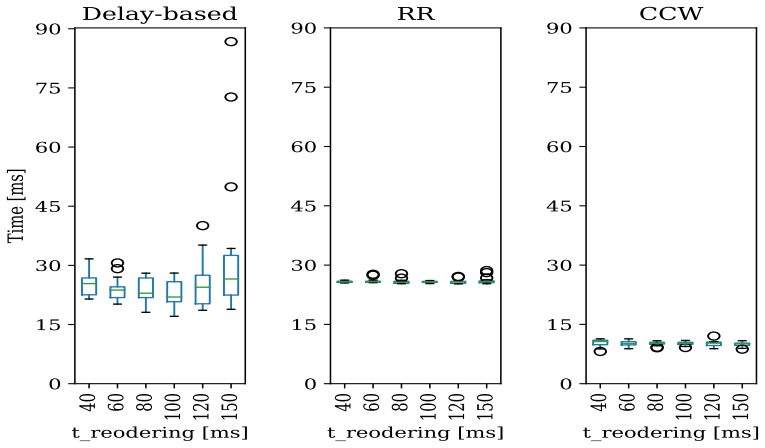
FIGURE 3.9: Average PDCP sojourn times for TCP traffic using different reordering timeout values.

Consequently, the CCW can achieve at least 95% of the ideal aggregate throughput regardless of the *t-Reordering* value and scenario. Nevertheless, Fig. 3.11 shows that the PDCP sojourn time is significantly higher than the one created with the Delay-based in both scenarios. Intuitively, this higher PDCP sojourn time would suggest a lower performance of the CCW. However, when the throughput in both BSs is significantly different, like in Scenario B, the CCW notoriously obtains a higher aggregate throughput than the Delay-based and RR algorithms.

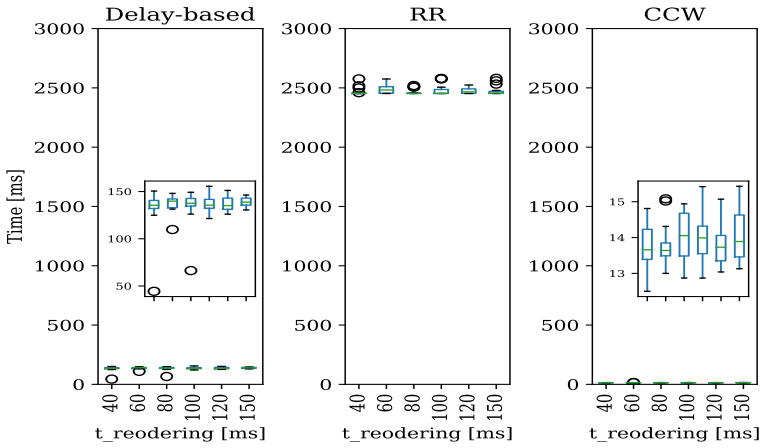
Furthermore, the results illustrated in Figs. 3.10c and 3.10d for scenario B indicate that the CCW keeps on average 4 ms of buffering delay difference between the RLC buffers. On the other hand, such delay for the Delay-based and RR is approximately 70 ms and 2500 ms, respectively, causing one communication path to be faster. Since the 3GPP reordering mechanism cannot deal with such delay differences, the performance of the algorithms mentioned above is significantly lower than the CCW. Because of this, the aggregate throughput obtained with the Delay-based and RR algorithms slightly increases using higher *t-Reordering* values. Note that the PDUs may



(A) MN's RLC delay in Scenario A



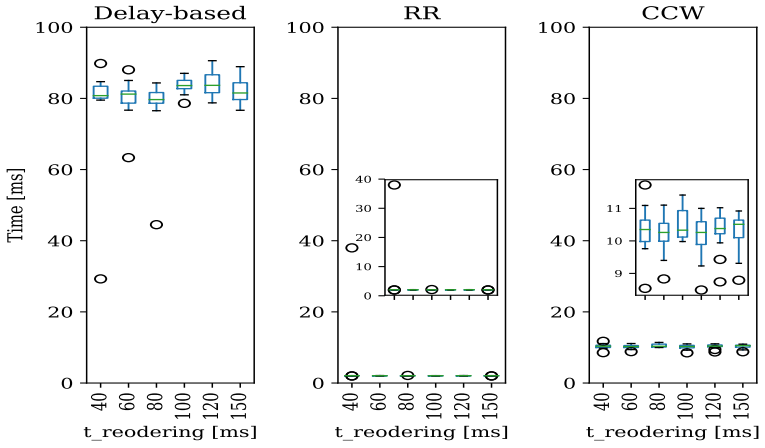
(B) SN's RLC delay in Scenario A



(C) MN's RLC delay in Scenario B

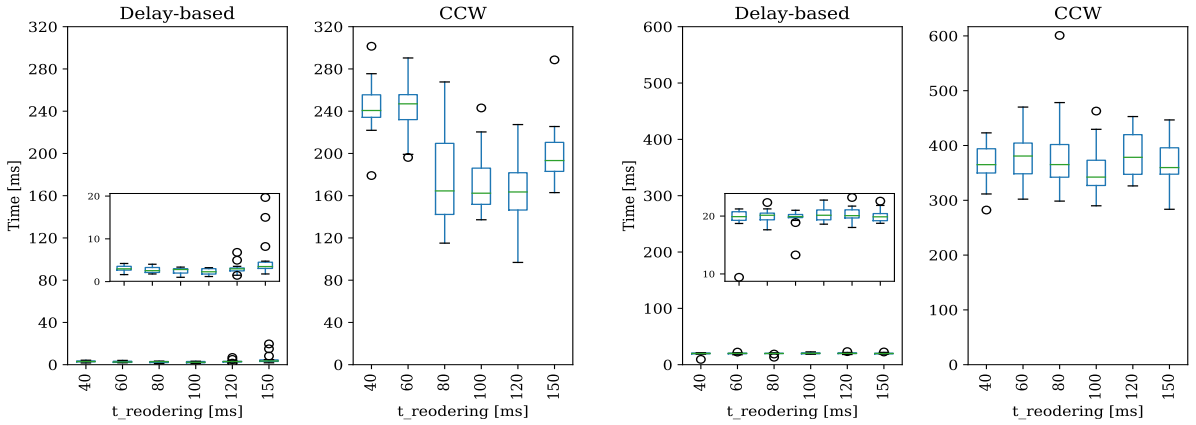
FIGURE 3.10: Average RLC sojourn times for UDP traffic using different reordering timeout values.

have to wait longer in the reordering buffer with higher reordering timeout values, which may cause a bufferbloat. It is important to remark that our CCW flow control algorithm aims not



(D) SN's RLC delay in Scenario B

FIGURE 3.10: Average RLC sojourn times for UDP traffic using different reordering timeout values (cont.)



(A) PDCP delay in Scenario A

(B) PDCP delay in Scenario B

FIGURE 3.11: Average PDCP sojourn times for UDP traffic using different reordering timeout values.

to reduce the end-to-end delay but to maximize the aggregate throughput, which is achieved in the evaluated scenarios for both TCP and UDP traffic.

3.5.3 The CCW Implementation Impact

This section evaluates the impact that the modification of two important parameters of the CCW flow control algorithm has on the obtained aggregate throughput. For this, we show in Figs. 3.12 and 3.13 the performance of the CCW in the scenario B for TCP and UDP traffic.

3.5.3.1 Splitting Time Interval

As we showed in Figs. 3.5 and 3.6, the CCW is the flow control algorithm that offers the best performance under the evaluated conditions/scenarios for UDP and TCP traffic. However, as mentioned in Section 3.3.3, the value chosen for the splitting time interval, i.e., T_{CCW}, may affect the efficiency of the CCW. In light of that, we depict in Fig. 3.12 the aggregate throughput for the CCW obtained for scenario B using a $T_{CCW} = 10$ ms. Note that this value doubles the one used to benchmark the CCW against the Delay-based and RR algorithms, the results of which are illustrated in Figs. 3.5 and 3.6.

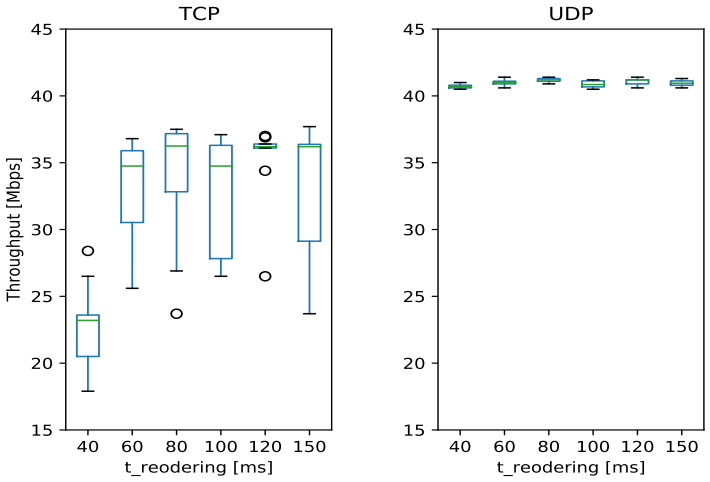


FIGURE 3.12: The effect of setting the T_{CCW} to 10 ms for scenario B and different reordering timeout values.

As we can appreciate in Fig. 3.12, using a larger T_{CCW} value reduces the obtained throughput for all the evaluated reordering timeout values. Indeed, compared to the throughput obtained with $T_{CCW} = 5$ ms, the reduction ranges from 22% to 45% for TCP traffic. Similarly, for UDP traffic, the performance diminishes by 22% approximately. The significant throughput reduction is caused by the slow response of the CCW to the variable radio link conditions that each BS faces. During the duration of the experiment, on one side, the RLC buffers may not have enough data to transmit at each TTI. On the other, the RLC buffering delay can be larger than the maximum delay set for the CCW, i.e., 20 ms. Here, the CCW splits the PDCP PDUs using only one BS until the buffering delay goes down. It is worth mentioning that since the MN and SN are separated by a backhaul with non-zero delay, having smaller values for T_{CCW}, i.e., smaller that 5 ms, does not necessarily improve the aggregate throughput. Indeed, this requires that the SN reports the CC_R more frequently, increasing the capacity requirements in the backhaul and the processing capacity at the MN. In this regard, having a T_{CCW} value equal to the TTI value would allow the BSs to have the exact amount of data that should be transmitted at each transmission opportunity. However, in a typical MR-DC deployment, the non-zero backhaul delay makes this a no viable solution. Therefore, setting the T_{CCW} to value close to the TTI allows the CCW to achieve a trade-off between high performance and low signaling requirements.

3.5.3.2 RLC Buffering Delay

One of the key aspects of the CCW flow control algorithm to beat the Delay-based and RR algorithms is to limit the maximum RLC buffering delay experienced at both BSs, i.e., $D_q^* max$.

By doing this, the CCW makes that two PDCP PDUs with consecutive sequence numbers split via the MN and SN in the same T_{CCW}, respectively, arrive at the receiver PDCP layer within a time of at most $BH + D_q^*max$. At a glance, the value chosen for the D_q^*max may affect the obtained aggregate throughput. However, since such a value is configured in both RLC buffers, the delay difference between PDUs with consecutive sequence numbers split in the same T_{CCW} is still $BH + D_q^*max$.

Fig. 3.13 shows that the aggregate throughput obtained using a $D_q^*max = 40$ ms is, on average, practically the same as the one using a $D_q^*max = 20$ ms, except when the *t_reordering* is 40 ms for TCP traffic. In this case, 40 ms is not enough to compensate for the delay difference between communication paths, mainly ruled by the RLC buffering and backhaul delays. Note that the $D_q^*max = 20$ ms is the value used for benchmarking purposes in this chapter, which is detailed in Section 3.4.1.

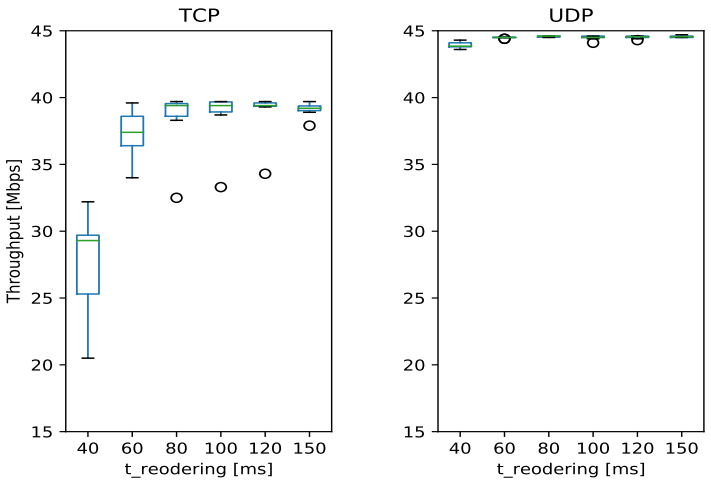


FIGURE 3.13: The effect of setting the D_q^*max to 40 ms for scenario B and different reordering timeout values.

It is worth mentioning that increasing the D_q^*max has not a significant impact on the overall PDU sojourn time since the PDUs are still buffered either at the PDCP or RLC buffers. Indeed, PDUs just move from one buffer to another. On the other hand, the sojourn delay is affected by the assigned radio resources, i.e., PRBs, and the packet losses that may arise due to the radio link conditions. In this regard, the D_q^*max should be set taking into account that a high D_q^*max value implies a higher PDCP reordering timeout value. Hence, if the D_q^*max is significantly large, the application may stop receiving data when PDCP PDU losses arise.

3.6 Summary

This chapter presented a flow control mechanism named CCW, which efficiently aggregates traffic from the MN and SN regardless of the MR-DC architecture option, MAC packet scheduler design, and transport layer protocol. The proposed algorithm utilizes the average size of the MAC SDUs and the average RLC buffering delay from both communication paths for the traffic splitting decision. We have extensively evaluated the performance of our proposal against the current state-of-the-art solutions using a mobile network testbed, which is built using the LTE/NR-complaint OAI software and software-defined radios.

The testbed experiments revealed that the out-of-order arrival of PDCP PDUs seriously degrades the performance of TCP traffic in MR-DC operation. However, if properly configured, the PDCP reordering mechanism alleviates this issue. Additionally, the benefit of aggregation is quantified in this study by an *aggregation benefit* metric, which indicates the efficiency of the adopted flow control algorithm to aggregate traffic from the MN and SN. In this regard, for TCP traffic, we showed that the proposed CCW algorithm achieves an aggregation benefit of more than 85%, where 100% means ideal aggregation, regardless of the reordering timeout choice and scenario. The RR algorithm achieves a similar performance only in Scenario A, i.e., when both BSs have the same bandwidth. Therefore, we demonstrated that the CCW outperforms the state-of-the-art Delay-based and benchmark RR flow control algorithms for TCP traffic in the evaluated scenarios and under the used parameters.

In addition, we showed that the performance of UDP-based traffic is not affected by the out-of-order arrival of PDUs. However, we noted that approximately half of the transmitted PDUs are received out-of-order regardless of the evaluated flow control mechanism. This condition can affect the perceived quality of some UDP-based applications. In this regard, we found that the CCW achieves an aggregation benefit of more than 95% regardless of the evaluated scenario. The Delay-based and RR algorithms achieve such performance only in Scenario A. For Scenario B, both algorithms offer a more inferior aggregation benefit.

Furthermore, the results demonstrated that maintaining an upper limit on the RLC buffering delay of both BSs makes the CCW obtain an aggregate throughput independent of the PDCP reordering timeout value. Indeed, in the evaluated setup, 60 ms is more than enough to compensate for the delay difference between communication paths. This CCW's feature facilitates the dimensioning of the PDCP reordering timeout value and avoids excessive buffering delay in the receiver PDCP that can increase the delay mentioned above.

Chapter 4

Minimizing the data interruption time at the higher layers

Mobility events and radio link failures, which may occur during the data aggregation, may pose challenges in meeting the latency, reliability, and throughput KPIs. Unlike SC, the UE in MR-DC operation can experience such events in either of the two BSs serving the UE with MR-DC. In typical MR-DC deployments, these events occur more frequent in the BS acting as the SN since the SN operates at a higher frequency band. In this chapter, we show that the data handovers and signal blockage events that occur at the SN can create out-of-order data reception or losses at the UE's PDCP layer, making the application stop receiving data for up to hundreds of milliseconds. Thus, challenging to meet the KPIs defined for such application. To mitigate this effect, we propose an intelligent and efficient mechanism that operates in the transmitting PDCP layer and significantly minimizes the data interruption periods suffered by the application when the UE aggregates data and HOs or failures of the SN occur.

Contributions:

- C. Pupiales, D. Laselva, and I. Demirkol, "Fast Data Recovery Mechanism for Improved Mobility Support in Multi-Radio Dual Connectivity," in IEEE Access, vol. 10, pp. 93 674–93 691, September 2021 [12]. (Area: Telecommunications; Rank: 43/94; Quartile: Q2; IF: 3.476).

4.1 Problem Description

In a typical MR-DC deployment, as shown in Fig. 4.1, one BS has macro cell coverage using frequencies in the range 1 (FR1), i.e., below 7.125 GHz, while the other BS has small cell coverage and may use FR2, i.e., 24.25 GHz to 52.6 GHz, [4, 7, 10]. In such common scenario, user mobility causes the link using the FR2 to suffer from signal blockages frequently, which might be caused either by loss in signal strength or because the UE transitions from the coverage area of one small cell to that of other small cell. In both cases, the UE losses connectivity with the corresponding BS. To recover the network connectivity, the UE or BS can perform a cell re-establishment in the former case, while the BS can trigger a HO for the latter case. Nevertheless, these procedures are time-consuming and cause the UE to stop receiving data from the small cell connection. The time the UE does not receive data is typically known as data

interruption time, representing, from the physical layer perspective, a few dozens of milliseconds [63]. However, from the perspective of the upper layers, i.e., transport and application layers, the data interruption time may be much higher because of the delay added by the MAC, RLC, and PDCP layers [33, 63]. In this chapter, we refer only to the data interruption time from the perspective of the upper layers, which is more critical for the applications.

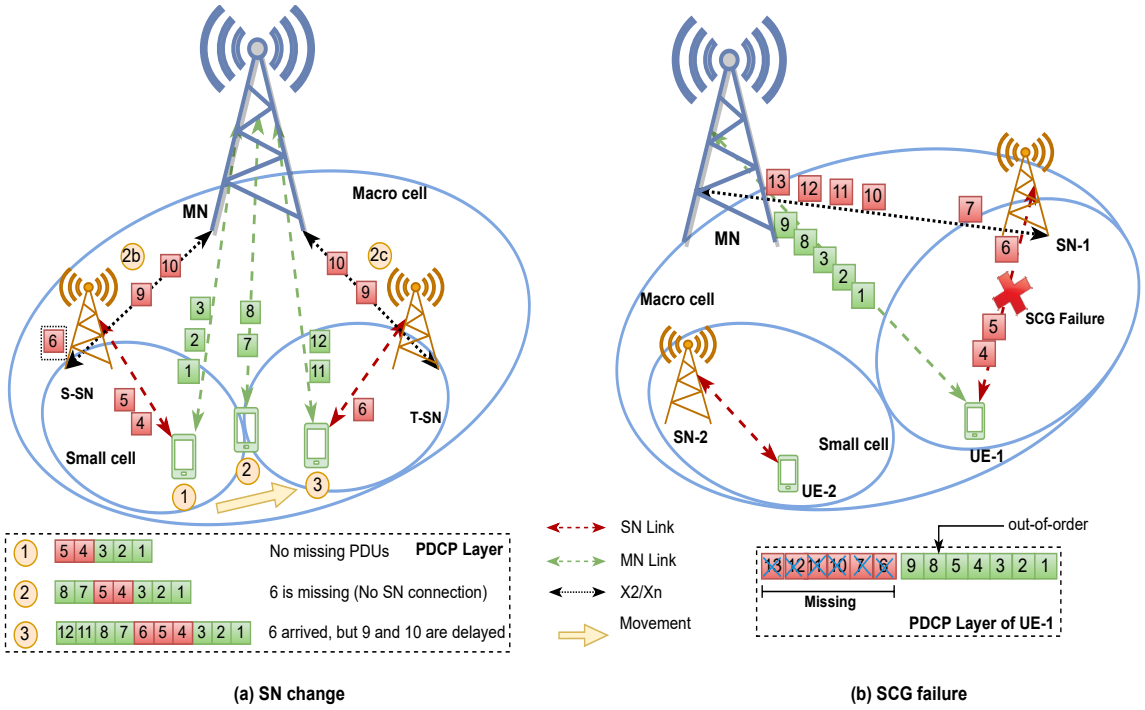


FIGURE 4.1: Out-of-order data reception caused by an SN change and SCG failure during data aggregation.

If the problems mentioned above occur while the UE aggregates data, the UE does not lose network connectivity completely. Indeed, it can still communicate via one of the BSs. Nonetheless, this can cause out-of-order PDCP PDUs arrivals, as illustrated in Fig. 4.1. In this scenario, the PDCP reordering mechanism [8] at the UE uses the PDU sequence number, a reordering timer, and a reordering window to wait for any delayed PDU to arrive and hence providing in-sequence data delivery to the upper layers. Despite this effective mechanism, an inadequate configuration of the reordering timeout value makes the PDCP layer discard the delayed PDUs too early or buffer the PDUs for an excessively long amount of time [9, 11]. In both cases, they are probably triggering the upper layer retransmission mechanisms. For instance, unlike UDP, the TCP receiver requests data retransmission when it detects sequence gaps, a.k.a., fast retransmission, or when the already transmitted data has not been acknowledged during a given period at the TCP sender, a.k.a., retransmission by timeout. In these cases, the TCP receiver stops delivering new information to the application layer until the missing data is correctly received, increasing the data interruption time to several hundreds of milliseconds. On top of that, the aggregate throughput is seriously affected since TCP reduces its congestion window. In this situation, meeting the KPIs defined for reliability- and latency-constrained applications such as low latency eMBB is challenging [64].

Despite the challenges that the data interruption time represent for the performance of MR-DC operation, the 3GPP has not defined any solution to tackle such a problem. Indeed, the SN change procedure [6], which is specified to manage the frequent changes of the SN, makes the UE stop communicating via the SN link until the change from the serving SN (S-SN) to

the target SN (T-SN) is completed, as depicted in Fig. 4.1(a). Moreover, the typical way to recover the network connectivity from a radio link failure (RLF), i.e., the blockage of the radio link, in SC operation is via a cell re-establishment procedure [31, 65]. However, when the SN link fails in MR-DC operation, a.k.a, secondary cell group (SCG) failure, such a solution is not specified by the 3GPP for MR-DC [31–33]. Therefore, the data buffered in the failing SN may be considered lost unless it can be transmitted to the UE using a new BS, as shown in Fig. 4.1(b). Unfortunately, this requires a new data forwarding procedure, which the 3GPP has not considered. Note that this scenario can also occur if the SN change procedure fails.

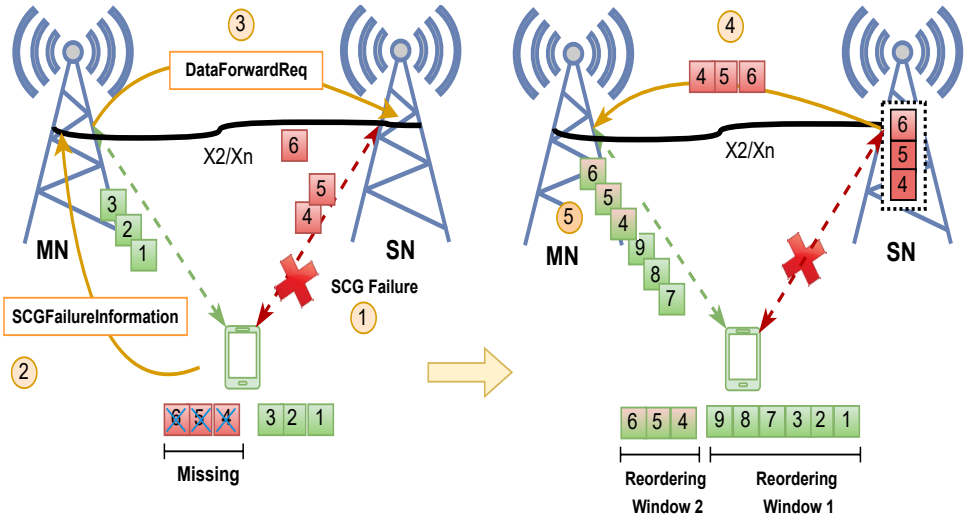


FIGURE 4.2: Data forwarding after an SCG failure event.

During SN-based HO or RLF, the out-of-order arrivals may be more frequent since the PDCP layer temporarily receives PDUs via only one BS. Having a large PDCP reordering timeout value can avoid discarding the delayed PDUs at the SN. Nevertheless, the application's data interruption time might also increase, making the application unable to achieve the required KPIs. For instance, low-latency eMBB applications such as VR and AR have a *motion-to-photon* latency requirement of less than 20 ms in order for the head's movement to match with the virtual scenery change [66]. In this regard, without considering the processing delays, their packet delay budget is around 7-15 ms [66]. If during the transmission of this kind of data, a HO is required or a RLF occurs, the latency requirement of less than 15 ms is almost impossible to achieve with the current SN change procedure, i.e., 74 ms as shown in Table 2.3 from Section 2.4.4.3. It is worth mentioning that it is impossible to use the DAPS HO in this case unless the UE has a third radio for MR-DC + DAPS.

Furthermore, a maximum aggregation benefit [58], i.e., the throughput gain obtained with MR-DC over SC, is achieved if the MN and SN maintain a continuous data flow with the UE[9]. For this, the RLC buffers of both BSs should have enough data to be transmitted at every transmission opportunity. In light of that, the flow control algorithm used in the MN should decide, based on each BS's radio link conditions and available radio resources, the number of PDCP PDUs to split via the MN and SN. Additionally, the different flow control algorithms available in the literature use up-to-date UE statistics, e.g., channel state information, from both BSs for the splitting decision making. However, during an SN change event, the UE is no longer connected to the corresponding SN; thus, there are no statistics to send back to the MN in order for the flow control mechanism to continue with the data splitting via the S-SN. A

reasonable action, in this case, can be to stop the data splitting via the S-SN until the UE has completed the RA with the new BS, i.e., the T-SN. However, this can cause the T-SN to have no data to transmit once the synchronization is completed, reducing the aggregate throughput. On the other hand, maintaining the data splitting via the S-SN can increase the out-of-order arrivals or losses of PDUs, exacerbating the application's data interruption time, especially if the SN change fails. In this regard, whether to suspend or not the data splitting via the S-SN, at which stage of the SN change procedure to do it, and when to resume the splitting, if stopped, are significant aspects to consider during an SN change event.

4.2 Related Works

Most of the available research efforts on MR-DC have focused on developing flow control solutions for data aggregation or methods to provide mobility robustness without data aggregation instead of reducing the application's data interruption experienced during HOs or signal blockages. For the former case, flow control mechanism mainly aim to maximize the user's aggregate throughput, reduce the end-to-end latency, maximize the throughput in one of the BSs, or achieve a minimum throughput for all users in both BSs [9, 39–43, 45, 46]. All theoretical and practical models presented in these works and their evaluations consider UEs without mobility and without service interruptions. Moreover, for the latter case, there have been studies that explore the use of MR-DC as an alternative to the legacy HO. In these studies, the authors state that MR-DC can reduce the HO failure probability, signaling exchange with the CN, the HO computational complexity, and HO completion delay [67–71]. Furthermore, these studies consider that before the HO, the UE already had CP connectivity via both BSs, i.e., the split bearer is configured in the MN and SN. However, the user's data is always transmitted via only one BS, i.e., the SN, upon triggering the HO. For this, the traffic is forwarded from the MN to the SN.

Furthermore, several studies show the capability of MR-DC to reduce the negative impact of link blockages on the performance of the user application's KPIs. For instance, in [72], the impact of various system parameters on the user's ergodic capacity for dense mmWave deployments is studied. Authors demonstrate that using multiple degrees of multi-connectivity, i.e., multiple radio link connections, helps to increase the achievable capacity by enabling backup connections. Additionally, in [73], authors indicate that having the UE with multi-connectivity reduces by up to seven times the denial of service and by up to ten times the dropping probability when static and dynamic blockages appear at the density of one blocker per square meter. Moreover, the theoretical framework presented in [74] suggests that under a high-density BSs deployment, extensive UE coverage, and short HO execution time, dual connectivity is sufficient to achieve the reliability target required in URLLC services in the presence of signal blockers. However, the multi-connectivity degree needed to support VR/AR services may be higher, especially in ultra-dense deployments. Authors in [75] state that blockages reduce the line of sight probability between the UE and BSs, implying that the UE has fewer available BSs to connect with in the area. This, in turn, increases the HO likelihood in BSs that use mmWaves. As a result, having multiple radio link connections improves the link reliability under simultaneous mobility and blockage events.

Our literature survey shows the need for studying solutions to minimize the data interruption periods that the application suffers during data aggregation when SN change or SCG failure events occur. This problem has been addressed only from the radio level perspective, i.e., the physical layer, and in SC scenarios. Nevertheless, since the splitting and aggregation processes

are done at the PDCP layer, an effective solution to the problem must include the perspectives of transport and application layers. Based on our knowledge, no work has studied the impact that mobility and signal blockages have on the performance of data aggregation with MR-DC, nor proposed solutions for such a problem. Actually, this topic has been included in the MR-DC enhancements plan for the future 3GPP Release 18 [64, 76–79]. For this reason, we propose a fast data recovery method for MR-DC that intelligently and effectively minimizes the application's data interruption time as described in the following sections.

4.3 FaRe Design Principles

Our *FaRe* mechanism aims to minimize the data interruption time that the application experiences during SN change or SCG failure events. The main functionality of the *FaRe* mechanism operates in the MN's PDCP layer, but similarly to the flow control, it also needs information from the SN. In light of this, the *FaRe* works along with a flow control algorithm to facilitate the data splitting management, i.e., stop/pause/resume the splitting, during SN change or blockage events.

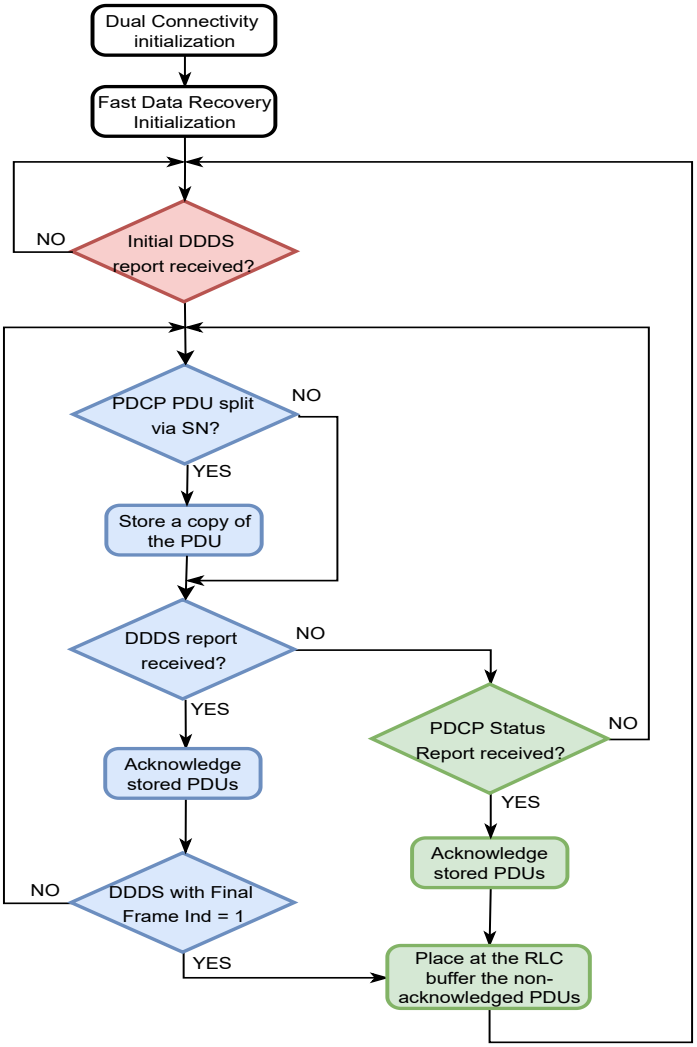


FIGURE 4.3: Flow chart of the functional stages of the *FaRe* mechanism at MN/anchor PDCP.

The *FaRe* has three main functional stages: the buffering stage, the fast retransmission stage, and the splitting activation stage. These functional stages work according to the flow chart depicted in Fig. 4.3, in which the blue, green, and pink colors represent the buffering, fast retransmission, and splitting activation stages, respectively.

4.3.1 Buffering Stage

In MR-DC operation, the SN cannot independently manage an SN change or SCG failure event. Actually, the MN makes the SN release/change/reconfigure decisions. For example, if an SCG failure occurs, the data buffered in the failing SN cannot be directly forwarded to a new BS. Instead, the MN has to request such a forwarding action, the data of which can be available for transmission in a new BS within $2*backhaul\_delay$ ms, assuming equal backhaul delays between BSs and zero message processing delay. Hence, depending on the backhaul delay, achieving a given application's latency target can be challenging.

To tackle this challenge, the *FaRe* dispenses with the data forwarding procedure. Instead, it temporarily stores a copy of the PDCP PDUs split via the SN in a newly defined buffer, named *FaRe-Buffer*, which is placed at the MN's PDCP layer, as depicted in Fig. 4.4. Note that the PDU copies are stored in a FIFO manner, being the head of the buffer, the PDU copy with the smallest PDCP sequence number. To reduce the memory requirements at the MN, a PDU is buffered until the SN indicates to the MN that such a PDU has been successfully delivered (RLC AM case) or transmitted to the UE (RLC UM case).

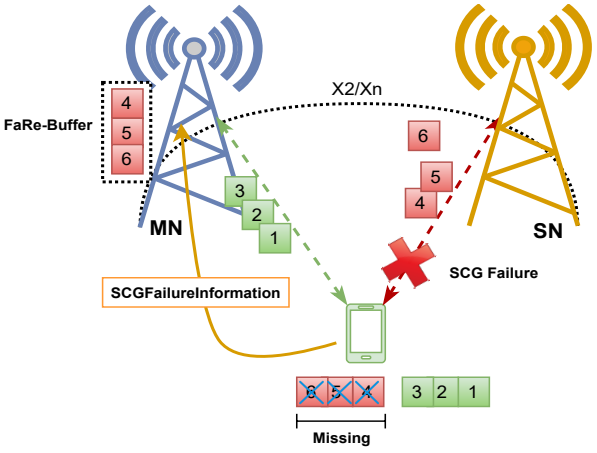


FIGURE 4.4: Buffering stage for recovery purposes of the *FaRe*.

To acknowledge the PDUs, the *FaRe* uses the information provided by the SN in the 3GPP-standardized Downlink Data Delivery status (DDDS) report [26], i.e., the highest transmitted/delivered PDCP sequence number. Once the PDUs' transmissions have been acknowledged, the corresponding *FaRe-PDUs*, i.e., the PDCP PDU copies present in the *FaRe-Buffer*, are deleted. It is worth mentioning that the SN can periodically send the DDDS report through the $Xn/X2$ interface, or the MN can explicitly request it via the same interface [26]. The Algorithm 3 describes the procedure performed by the MN to store and acknowledge the *FaRe-PDUs*.

Algorithm 3 FaRe Buffering Stage Algorithm

Input: PDCP PDU or DDDS report
Output: *FaRe-PDUs*, Acknowledged *FaRe-PDUs*

```
1: while FaRe is enabled do
2:     if PDCP PDU is forwarded to the SN then
3:         Make a copy of the PDU
4:         Place the copy in the FaRe-Buffer
5:     else
6:         Continue
7:     if DDDS Type 1 is received then
8:         ACK_SN = highest transmitted/delivered SN
9:         while FaRe-PDU ≤ ACK_SN do
10:            Delete the FaRe-PDU
11:    else
12:        Continue
```

4.3.2 Fast Retransmission Stage

Upon triggering an SN change or SCG failure, the fast retransmission stage retransmits to the UE the PDUs copies buffered at the *FaRe-Buffer*, as shown in Fig. 4.5. For this, it can use the information provided within the DDDS report or SCGFailureInformation message [31].

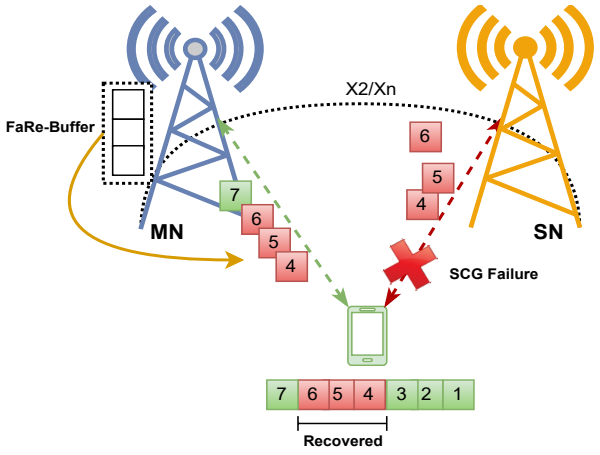


FIGURE 4.5: PDCP-level retransmission stage of the *FaRe*.

4.3.2.1 SN Change Scenario

As we mentioned earlier, the MN should stop the data splitting via the SN during an SN change event to minimize the out-of-order arrivals, thus the application's data interruption time. In this regard, the *FaRe* communicates the flow control mechanism to stop the data splitting via the SN before the MN sends either the *SN Release Request* or *SN Change Confirm* messages. Actually, the MN can even stop the data splitting earlier if the radio link conditions experienced between the UE and S-SN are not favorable to maintain the connectivity.

Upon receiving the *SN Release Request* or *SN Change Confirm* messages, the S-SN stops communicating with the UE and releases all the radio resources assigned to the corresponding UE

[6]. For this reason, the S-SN prepares and sends through the $X2/Xn$ interface a DDDS report that includes the latest delivered/transmitted PDCP PDUs and the indication that this report is the final one, i.e., the Final Frame Indication flag is activated. After receiving the final DDDS report, the MN acknowledges the corresponding *FaRe-PDUs* and updates the *FaRe-Buffer* accordingly. Afterward, all the remaining *FaRe-PDUs* are placed at the head of the RLC buffer to be transmitted in the next transmission opportunity. Consequently, these PDUs can rapidly be delivered to the UE using the MN's Uu instead of waiting for the completion of the SN change procedure. Note that if the RLC buffer contains a segmented SDU, the *FaRe-PDUs* are placed after this SDU.

Considering the SN change procedure initiated by the SN as described in [6], a PDCP PDU that was ready to be transmitted by the S-SN but which was forwarded to the T-SN can arrive at the UE's PDCP in approximately 74 ms with the current SN change procedure, as shown in Table 2.3. However, with the *FaRe* mechanism, the same PDU can arrive with a delay computed as

$$\Delta_{FaRe} = BH + UP_{delay} + \Delta_{proc} + n \times TTI, \tag{4.1}$$

where Δ_{FaRe} is the elapsed time from the S-SN stopping the communication with the UE until the first *FaRe-PDU* arriving at the UE's PDCP, BH is the backhaul delay, UP_{delay} is the user plane latency, e.g., 4 ms for eMBB [28], Δ_{proc} is the time that takes to process the DDDS report and place the *FaRe-PDUs* in the RLC buffer, n is the number of elapsed TTIs until the first *FaRe-PDU* is scheduled for transmission, and TTI is the Transmission Time Interval, e.g., 1 ms for eMBB. Therefore, assuming $BH = 5$ ms, $UP_{delay} = 4$ ms, Δ_{proc} as negligible, $TTI = 1$ ms, and $n = 1$, i.e., the PDU is scheduled in the next TTI, the Δ_{FaRe} is 10 ms. This delay is significantly lower than the one achieved with the current SN change procedure, i.e., 74 ms.

4.3.2.2 SCG Failure Case

Upon receiving the *SCGFailureInformation* message from the UE, the *FaRe* communicates the flow control mechanism to stop the data splitting via the SN. Afterward, based on the PDCP Status Report [8] included by the UE in the *SCGFailureInformation*, the PDUs that have not been received at the UE are placed at the head of the RLC buffer so that they can be transmitted in the next transmission opportunity. The novel inclusion of the PDCP Status Report in the *SCGFailureInformation* message helps the MN to recover those PDUs that were erroneously received at the UE, which is common during poor radio link conditions like those experienced before declaring an RLF. The elapsed time from the MN receiving the *SCGFailureInformation* message until the UE's PDCP receiving the first *FaRe-PDU* is denoted as Δ^*_{FaRe} and computed as

$$\Delta^*_{FaRe} = UP_{delay} + \Delta_{proc} + n \times TTI. \tag{4.2}$$

Assuming $UP_{delay} = 4$ ms, $FaRe_{proc}$ as negligible, $TTI = 1$ ms, and $n = 1$, the Δ^*_{FaRe} is approximately 5 ms.

It is worth mentioning that there is no recovery mechanism in the literature that deal with the SCG failure problem. However, a possible approach could be to request data forwarding from the failing SN. In this case, the time that may take for the first non-delivered PDU to arrive at the UE's PDCP is denoted as DF and computed as

$$DF = 2 \times BH + UP_{delay} + RLC_{delay} + DF_{proc}, \tag{4.3}$$

where RLC_{delay} is the RLC buffering delay experienced at the moment that the forwarded PDUs ingress to the RLC buffer and DF_{proc} is the processing delay. Assuming $BH = 5$ ms, $UP_{delay} = 4$ ms, DF_{proc} as negligible, and an RLC_{delay} of 10 ms, the DF is approximately 24 ms. Note that the RLC_{delay} will vary depending on the flow control algorithm in use. For instance, the CCW flow control algorithm limits the RLC buffering delay to 20 ms [9]. Hence, assuming an RLC_{delay} at the 50% of its limit, i.e., 10 ms, is a conservative approximation to the reality.

The *Fast Retransmission Stage* functionality of the *FaRe* is presented in Algorithm 4 and is described in the following.

Algorithm 4 FaRe: Fast Retransmission Stage

Input: *SN Addition Request Acknowledge or SCGFailureInformation* messages
Output: *FaRe-PDUs* delivered to the UE's PDCP

```
1:  while FaRe is enabled do
2:      if SN Addition Request Acknowledge received then
3:          Notifies the flow control to stop data splitting via SN
4:          while DDDS with FinalFrameInd = 0 do
5:              ACK_SN = highest transmitted/delivered PDU
6:              while FaRe-PDU ≤ ACK_SN do
7:                  Acknowledge FaRe-PDUs
8:                  Update the FaRe-Buffer
9:          if Head's RLC SDU is segmented then
10:             Place the FaRe-PDUs after the segmented RLC SDU
11:         else
12:             Place the FaRe-PDUs before the RLC SDUs
13:         Flush the FaRe-Buffer
14:     else if SCGFailureInformation received then
15:         Notifies the flow control to stop data splitting via SN
16:         Read the PDCP Status Report
17:         ACK_SN = First Missing PDU
18:         while FaRe-PDU < ACK_SN do
19:             Acknowledge FaRe-PDUs except the non-received PDUs included in the bitmap
20:             Update the FaRe-Buffer
21:         if Head's RLC SDU is segmented then
22:             Place the FaRe-PDUs after the segmented RLC SDU
23:         else
24:             Place the FaRe-PDUs before the RLC SDUs
25:         Flush the FaRe-Buffer
26:     else if Initial DDDS received then
27:         Notifies the flow control to resume data splitting via SN
28:     else
29:         Keep acknowledging FaRe-PDUs
```

- The MN stops splitting PDCP PDUs via the BS acting as SN when it receives either the *SN Addition Request Acknowledge* or *SCGFailureInformation* message (Lines 2-3 and 14-15, respectively).

- After receiving the *SN Addition Request Acknowledge* message, the MN waits for the DDDS report with $FinalFrameInd = 1$. Meanwhile, it acknowledges the *FaRe-PDUs* and updates the *FaRe-Buffer* with every received DDDS report (Lines 4-8).
- Once the DDDS report with $FinalFrameInd = 1$ is received, the MN places at the head of its RLC buffer all the *FaRe-PDUs* that were not acknowledged. Subsequently, the MN flushes the *FaRe-Buffer* (Lines 9-13).
- After receiving the *SCGFailureInformation* message, the MN reads the PDCP-related information, acknowledges the *FaRe-PDUs*, and updates the *FaRe-Buffer*. If such a information includes the non-received PDUs, the corresponding *FaRe-PDUs* are not deleted from the *FaRe-Buffer*. Afterward, the *FaRe-PDUs* that were not acknowledged are placed at the head of the RLC buffer. Finally, the MN flushes the *FaRe-Buffer* (Lines 17-28).
- The MN resumes the data splitting via the SN once it receives an initial DDDS report from the T-SN or a new BS acting as SN (Lines 26-27).

4.3.3 Splitting Activation Stage

As indicated in the Section 4.3.2, the *FaRe* makes the flow control to stop forwarding PDCP PDUs via the SN to avoid out-of-order deliveries, thus a higher data interruption time for the application. To avoid the sharp drop in the aggregate throughput during the SN change, the T-SN should start transmitting the user data to the UE as soon as the RA procedure is completed. Nevertheless, having the T-SN's RLC buffer with sufficient data to be transmitted in the next transmission opportunity without increasing the data interruption time is a challenge. For instance, if the SN change procedure fails, the PDCP PDUs present in the T-SN's RLC buffer should be forwarded to a new T-SN. Nevertheless, this time-consuming procedure may cause the aforesaid PDUs to arrive at the UE after the PDCP reordering timer expires. Hence, as they no longer belong to the reordering window, they are discarded [8], increasing the application's interruption time, especially for TCP-based traffic.

Since the *FaRe* quickly transmits the non-delivered PDUs via the MN's Uu and makes the flow control to operate in SC-like mode during the SN change event, the receiving PDCP layer never stops sending data to the upper layers. Therefore, the throughput never goes down to zero. For this reason, the MN only resumes the data splitting via the SN when it receives from the T-SN the initial DDDS report [26], which indicates that the UE has completed the RA procedure and it is ready to receive new data. This procedure is illustrated in Fig. 4.6.

Due to the T-SN's RLC buffer is empty, it cannot send to the MN any flow control statistic in order for the flow control algorithm to compute the number of PDCP PDUs to split via the SN. To overcome this challenge, the *FaRe* indicates the initial values for the variables that the flow control algorithm uses for its splitting decisions. For instance, the CCW flow control algorithm uses the MAC SDU sizes and RLC buffering delay statistics for its data splitting decisions [9]. Similarly, the Delay-based flow control algorithm uses the PDCP PDU transmission delay computed from the UE status report [46]. In this regard, considering these two state-of-the-art flow control algorithms, the above mentioned initial values are given by

4.3.3.1 The CCW Case

$$SDU_{DC}^{initial} = \min(SDU_{DC}^{1}, ..., SDU_{DC}^{k}), \tag{4.4}$$

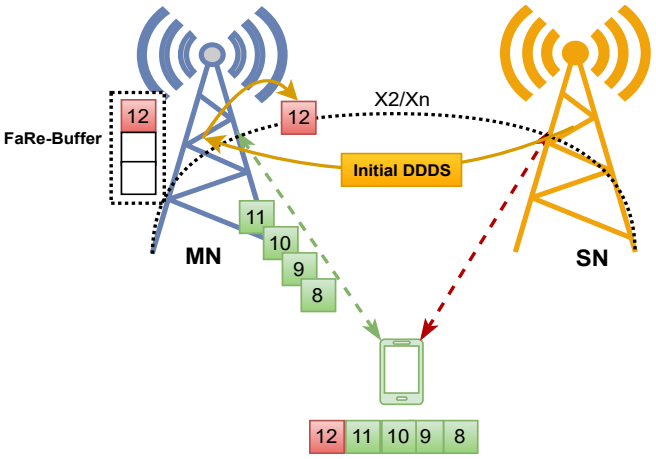


FIGURE 4.6: Splitting activation stage of the *FaRe*.

$$RLC_{delay} = 0, \tag{4.5}$$

where $SDU_{DC}^{initial}$ is the initial value to use for the MAC SDU size variable, SDU_{DC}^{1} is the first MAC SDU size statistics received from the S-SN right after the MN sends the *SN Change Required* or *SN Addition Request* message, SDU_{DC}^{k} is the last MAC SDU size statistics received from the S-SN just before the MN sends the *SN Change Confirm* or *SN Release Request* message, and RLC_{delay} is the initial value to use for the RLC buffering delay. It is worth mentioning that during a HO, the quality of the radio link conditions does not allow the UE to achieve high data rates. However, in MR-DC operation, the BSs must assure a minimum of radio resources for the UE to perform a given minimum data rate. For this reason, the *FaRe* uses the minimum MAC SDU size statistics received during the period mentioned above as the initial value for the MAC SDU size variable. Likewise, since the T-SN's RLC buffer is empty after the RA procedure, the *FaRe* indicates the CCW to use 0 ms as the initial value for the RLC buffering delay variable.

4.3.3.2 The Delay-based Case

$$PDU_{delay} = 0, \tag{4.6}$$

where PDU_{delay} is the PDCP PDU transmission delay. Since during and right after the completion of the RA procedure, the T-SN has no data in its RLC buffer, the *FaRe* indicates the Delay-based to use 0 ms as the initial value for the PDCP PDU transmission delay.

Regardless of the flow control algorithm used in the MN to split the incoming data, the *FaRe* indicates the flow control algorithm to use the initial values for their variables until the MN receives up-to-date statistics from the T-SN. Likewise, the *FaRe's Buffering Stage* is initiated as soon as the incoming traffic is split via the SN link. Note that after an SCG failure event, the UE switches to SC operation. However, the UE may recover the MR-DC operation, thus the data aggregation if the MN initiates the procedure to re-establish the connection with the failing SN [31]. On the contrary, if the MN decides to add a new SN, the data aggregation is started from scratch. Thus, the *Traffic Activation Stage* is not applicable in this case. The splitting activation stage is presented in the Algorithm 5.

Algorithm 5 FaRe: Splitting Activation Stage

Input: *SN Addition Request or SN Change Required* messages
Output: Initial values for the flow control variables

while FaRe enabled **do**
  if *SN Addition Request* **or** *SN Change Required* sent **then**
    if CCW enabled **then**
      Set $SDU_{DC}^{initial} = 0$
      Set $RLC_{delay} = 0$
      while *SN Release Request* **or** *SN Change Confirm* has not been sent **do**
        if Flow control statistics received **then**
          if $SDU_{DC}^{initial} \leq SDU_{DC}$ **then**
            $SDU_{DC}^{initial} = SDU_{DC}$
          else
            Maintain the old $SDU_{DC}^{initial}$
    else if Delay-based enabled **then**
      Set $PDU_{delay} = 0$
    else
      Set the flow control variables accordingly.
  if *initial DDDS* report received **then**
    if CCW enabled **then**
      Set $SDU_{DC} = SDU_{DC}^{initial}$
      Set $D_q = 0$
    else if Delay-based enabled **then**
      Set $PDU_{delay} = 0$
    else
      Set the flow control variables accordingly.
    Indicates the flow control to resume the splitting via the SN

4.4 Evaluation Framework

To validate our proposed *Fast Data Recovery* mechanism for MR-DC, we use the Dual Connectivity [5] solution implemented on the LTE/NR testbed developed using the OAI [34], which is widely detailed in Section 2.5.1. The testbed is based on the split DRB architecture and implements the user plane functionalities of DC detailed in [8]. The testbed setup is shown in Table 4.1 and the evaluation scenarios are given in the following.

TABLE 4.1: General configuration for the BSs

Parameter	Value
Duplex Mode	FDD
E-UTRA Band	7
DL Frequency for MN	2.68 GHz
DL Frequency for SN	2.63 GHz
Bandwidth for MN	5 MHz
Bandwidth for SN	10 MHz
RLC mode	UM

4.4.1 Benchmarking Strategies

As indicated in Section 4.1, the problem of the data interruption at the application level caused by SN-based HO/RLF while the UE aggregates data has not been addressed yet in the available literature. Therefore, to validate the performance of the proposed *FaRe* mechanism, we have implemented different possible strategies to follow in case an SN change is required or an SCG failure occurs. For both cases, the baseline strategies use the state-of-the-art CCW flow control algorithm [9]. The CCW dynamically splits the incoming UP traffic via the MN and SN according to the average MAC SDU sizes and RLC buffering delays experienced in both BSs. To maximize the average aggregate throughput and achieve a stable value, the CCW maintains the RLC buffers of both BSs with enough data to prevent link starvation while avoiding congestion [9].

4.4.1.1 SN Change Scenario

In the first strategy, *Baseline_1*, the MN, just before sending the *SN Change Confirm or SN Addition Request* messages, resets the flow control variables related to the MN and S-SN, and makes the flow control operate under "initial conditions" until up-to-date statistics are received from both BSs. Operating under "initial conditions" implies for the CCW to split the data in a Round-Robin manner until the RLC buffering delay and MAC SDU size statistics from both BSs are known [9]. Note that the MN never stops splitting the incoming UP traffic via both BSs.

For the second strategy, *Baseline_2*, the MN stops the data splitting via the S-SN before sending the *SN Change Confirm or SN Addition Request* messages. The splitting is resumed when the MN receives the "initial" DDDS report from the T-SN, which indicates that the UE is successfully synchronized. Note that the PDUs that were not transmitted by the S-SN are forwarded to the T-SN as defined by 3GPP in [6]. Since the MN resumes the splitting only if the UE is synchronized with the T-SN, possible data losses or out-of-order deliveries caused by RA or RRC reconfiguration failures are avoided.

Finally, in the third strategy, *Baseline_3*, the MN resets the flow control variables related to the S-SN and suspends the data splitting via the S-SN before sending the *SN Change Confirm or SN Addition Request* messages. The MN resumes the data splitting when it receives up-to-date statistics from the T-SN. If the S-SN has no data to forward to the T-SN, the latter has no data to transmit to the UE once the RA procedure is completed. Thus, the MN will not resume the data splitting.

4.4.1.2 SCG Failure Scenario

For the baseline strategy, *Forward_Req*, upon receiving the *SCGFailureInformation* message, the MN requests the failing SN to forward the non-transmitted/delivered PDCP PDUs so that they can be transmitted via the MN's Uu. When the requested PDUs arrive at the MN, they are transmitted after the RLC SDUs present in the RLC buffer. Moreover, since during a data session the UE may experience SN change events before the SCG failure event occur, the *Forward_Req* will use the *Baseline_2* strategy to manage the HO events. This corresponds to a default implementation that follows the 3GPP specifications [6, 31].

4.4.2 Performance Metrics

To evaluate the performance of our *Fast Data Recovery* mechanism on reducing the application's data interruption time during SN change and SCG failure events, we use the following metrics:

4.4.2.1 Aggregate Throughput

We evaluate the aggregate throughput obtained at the transport layer with every Baseline strategy and the *FaRe* after a data session of 30 seconds for the SN change case, and 20 seconds for the SCG failure case. In this regard, we evaluate how the SN changes and SCG failure events impact on the instantaneous and average aggregate throughputs. Additionally, for a better illustration of the short-term impact of aforementioned events, the obtained aggregate throughput samples are collected every 100 ms.

4.4.2.2 Throughput Variance

To evince and compare the variability of the throughput caused by SN change and SCG failure events in the entire data session, we use the Variance Ratio (R_{var})[67], which is defined as

$$R_{var} = \frac{\delta_{T_{DC}}}{T_{DC}}, \tag{4.7}$$

where T_{DC} is the average aggregate throughput obtained by the application at the end of an experiment and $\delta_{T_{DC}}$ is the standard deviation of T_{DC}. Note that high values of R_{var} indicate significant throughput instability, such as long periods of zero throughput or short periods with very high throughput peaks.

4.4.2.3 Data Reliability

When a UE aggregates data, the main goal is to maximize the obtained throughput. However, achieving a given reliability target while maximizing the throughput may be challenging for some applications during SN change or SCG failure events. In this regard, we evaluate the reliability obtained at the PDCP level with and without the use of the *FaRe* mechanism. For this, we compare the number of PDCP PDUs that leave the MN's PDCP layer with the PDCP PDUs received in the mirroring layer during the entire data session. The PDCP reliability (R_{PDCP}) is defined as

$$R_{PDCP} = \frac{PDUs^{RX}}{PDUs^{TX}} \times 100\%, \tag{4.8}$$

where $PDUs^{RX}$ is the number of PDCP PDUs that are successfully received at the UE, and $PDUs^{TX}$ is the number of PDCP PDUs that are split by the MN and leave the PDCP layer to be transmitted via either BS.

4.4.2.4 Data Interruption Time

When the UE aggregates data, the interruption time experienced at the transport and/or application layers is influenced by the out-of-order arrival of PDCP PDUs or PDCP PDU losses.

In this regard, the interruption time increases while the PDUs spend more time in the PDCP reordering buffer. Unlike UDP, TCP is a reliability-oriented protocol, so it must provide in-sequence delivery to the application. Therefore, if TCP sequence gaps are detected, the application will not receive data until the lost packet is correctly recovered by TCP. For this reason, we measure the elapsed time the transport layer stops receiving data during the data session. Note that *iperf3* measures the throughput at the transport layer, but these results also represent the data interruption time at the application level.

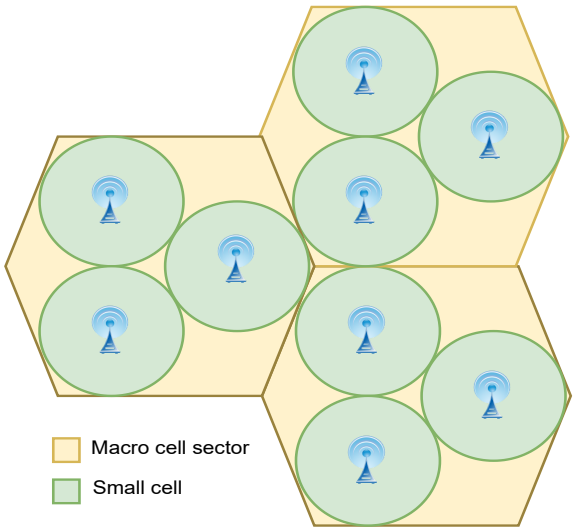
4.4.3 Evaluation Scenario

To recreate a mobility scenario on the DC testbed described in Section 2.5.1, we use the SINR, CQI, and reference signal received power (RSRP) traces extracted from the Nokia-proprietary system-level simulator for the MN and SN using a 3GPP-defined DC scenario detailed in [80]. Note that the system-level simulator is entirely in alignment with 3GPP simulations and modeling guidelines. In addition, the SINR values from the aforementioned traces are measured every 5 ms. In this regard, each time a CQI is to be reported by the OAI UE, the next trace value is used. Hence, each OAI BS independently receives from the OAI UE, i,e., mUE and sUE stacks in our setup, the CQI value using the CSI report every 5 ms.

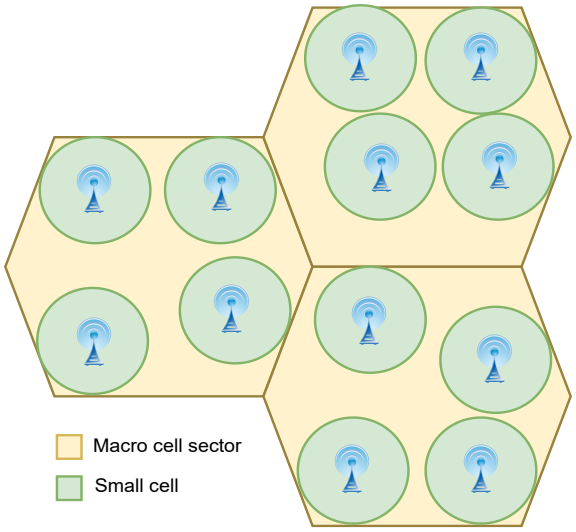
To evaluate intra-MN MR-DC HO scenarios, we define two 3GPP-defined DC deployment scenarios. In the first one, named *Dense Deployment*, the macro cells have an inter-site distance (ISD) of 200 meters with nine small cells uniformly distributed across the macro coverage area, i.e., each macro sector has three small cells. This deployment may create coverage gaps of less than 50 meters between small cells, as described in [80] and depicted in Fig. 4.7a. In the second deployment scenario, named *Sparse Deployment*, the macro cells have an ISD of 500 meters with clustered small cell groups, deployment like scenario 2a detailed in [81], i.e., each macro sector has four small cells. Due to the clustering and higher ISD, there can be about 200 meters of coverage gaps between small cells, as illustrated in Fig. 4.7b.

Additionally, in the scenarios described above, the UE never leaves the macro cell coverage area for our evaluations, i.e., the UE is always connected to the same MN. In such a scenarios, the UE moves at 30 Km/h following a single trajectory across the different small cells deployed in the macro cell coverage area. At the same time, the network is loaded with stationary UEs to create interference to have a realistic SINR. Furthermore, a 3 dB offset is considered for the HO decision, i.e., the A3 event, thus avoiding ping pongs between neighbor cells. Under the above-indicated conditions, during a data session of 30 seconds, the UE experiences nine SN change events in the dense deployment scenario and fourteen in the sparse deployment scenario. It is worth mentioning that the radio link conditions experienced by the UE with each BSs are not the same in both deployment scenarios. Indeed, the obtained aggregate throughput would be different.

In a typical MR-DC deployment, higher data rates in the small cell are achieved than in the macro cell thanks to higher bandwidth values assigned to the small cells. Therefore, we define a scenario where the MN and SN use different channel bandwidths, i.e., 5 MHz and 10 MHz, respectively. We only use this combination of bandwidth values because the OAI UE does not achieve a stable throughput with different values, e.g., 20 MHz [61]. In addition, since in our testbed, a single UE is connected to both BSs, all the physical resource blocks (PRBs) available for each channel bandwidth are assigned to the UE, i.e., 25 PRBs for the MN and 50 PRBs for the SN. In this scenario, we evaluate the performance of user traffic, which uses the TCP as transport layer protocol, during SN change and SCG failure events using the metrics described in



(A) Dense deployment.



(B) Sparse deployment.

FIGURE 4.7: Deployment scenarios.

Section 4.4.2 and the benchmarking strategies described in Section 4.4.1. Additionally, since the data interruption time may increase with the value configured for the PDCP reordering timer (t-Reordering), the TCP traffic is also evaluated using different reordering timeout values. For the evaluations, the downlink TCP traffic is generated using the *iperf3* tool, in which the *server* runs at the mUE's host, where the UE's PDCP layer resides, and the *client* at the EPC's host. Note that the *iperf3* intents to saturate the link to measure the maximum achievable throughput. Table 4.2 summarizes the scenario setup and provides additional configuration details.

4.5 Results and Discussion

This section shows and discuses in detail the results obtained using the framework presented in Section 4.4. According to the OAI specifications, the downlink throughput achieved for LTE

TABLE 4.2: Scenario evaluated summary.

Parameter	Value
Single experiment duration	30 sec for SN change, 20 sec for SCG failure
Channel Bandwidth	$MN_{BW} = 5$ MHz $SN_{BW} = 10$ MHz
PDCP t-Reordering	100, 150, 200, 300 ms
PDCP reordering window	2048 PDUs
One-way backhaul latency	5 ms [29, 60]
Traffic Type	Saturated TCP
MR-DC architecture option	LTE-DC
Deployment scenarios	Dense, sparse
CCW configuration	$T_{CCW} = 5$ms, $D_q^*max = 20$ ms, and $\alpha = 0.3$ [9]

in SC operation under an ideal scenario, i.e., highest CQI value and no packet losses, is 16-17 Mbps with a channel of 5 MHz and 34-35 Mbps with a channel of 10 MHz [61]. However, as we mentioned in Section 4.4.3, the CQI values for the UE-MN and UE-SN connections vary depending on the UE's location with respect to the BS in the macro cell and small cells, based on the traces collected.

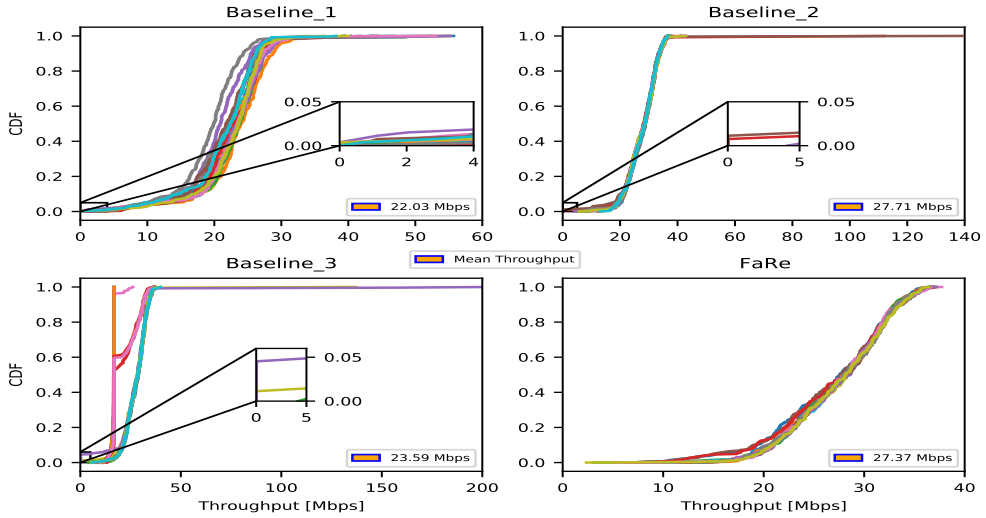
4.5.1 Aggregate Throughput

Contrary to the throughput obtained in a static and/or ideal scenario, i.e., no losses and highest CQI, the aggregate throughput obtained in a mobility scenario is affected by the number of HO events, radio link conditions' variability, and the approach used to minimize the application's data interruption time. To illustrate the variability of the aggregate throughput, we present the results using the cumulative distribution function (CDF) obtained for 20 different experiments, with throughput samples collected per experiment every 100 ms for a total duration of 30 seconds in the SN change scenario and 20 seconds in the SCG failure scenario. The results presented in Figs. 4.8 and 4.11 do not include the throughput values obtained during the TCP' slow start stage, i.e., the first 3.5 seconds of the experiment duration in our case, to avoid confusing them with those obtained during SN change and SCG failure events. It is worth mentioning that random events such as HARQ retransmissions or packet losses may appear in our LTE/NR testbed, creating differences in the throughput values obtained across the experiments.

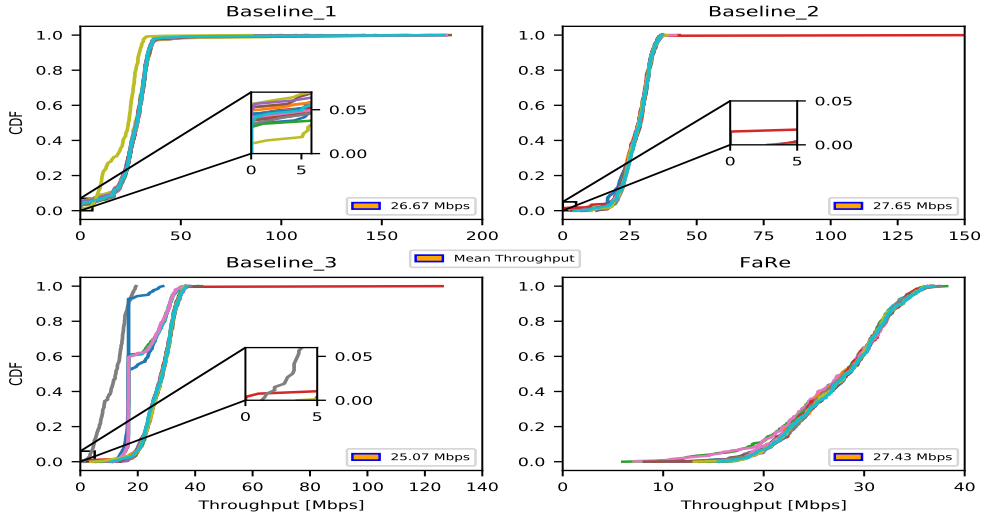
Furthermore, in this subsection, we evaluate the performance of the *FaRe* mechanism against the Baseline strategies described in Section 4.4.1. Fig. 4.8 and 4.10 illustrates the CDF of the aggregate throughputs obtained using the *t-Reordering* values detailed in Table 4.2, i.e., 100, 150, 200, and 300 ms, which help us to visualize the behaviour of the aggregate throughput based on different configurations for the dense and sparse deployments.

4.5.1.1 SN Change Scenario with a Dense Deployment

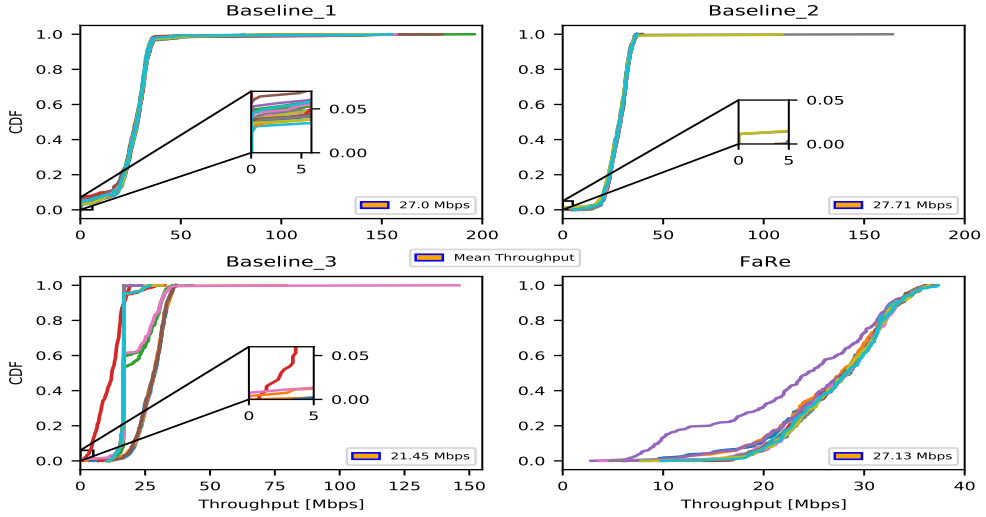
The *Baseline_1* strategy achieves the worse performance among the compared methods. In this case, Fig. 4.8 shows that the probability of having zero throughput values is more significant than in the other cases. Indeed, the probability is around 1% only when the *t-Reordering* is 100 ms. In the other cases, the probability goes up to 4-6%. As a consequence of the periods with zero throughputs, the *Baseline_1* reaches high peak values of approximately 180 Mbps, which



(A) CDF throughput for t-Reordering = 100 ms

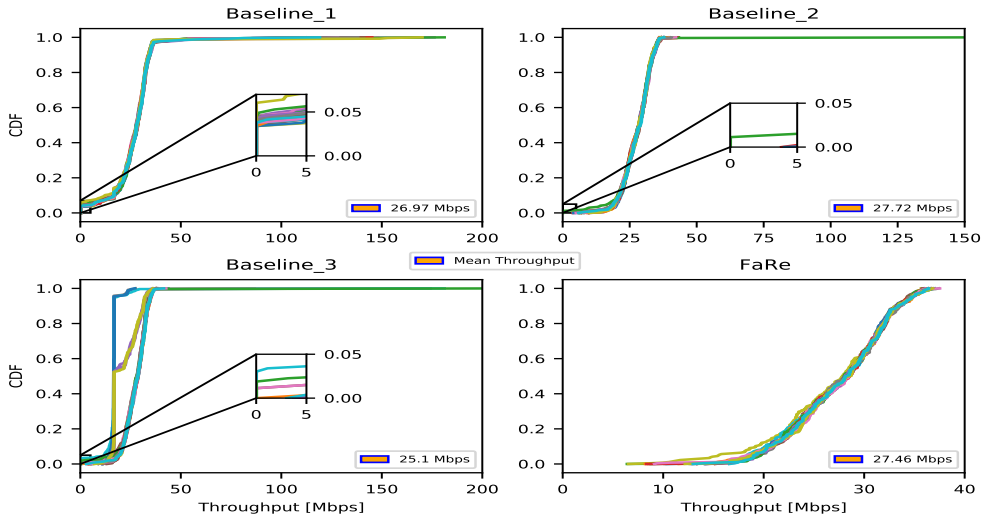


(B) CDF throughput for t-Reordering = 150 ms



(C) CDF throughput for t-Reordering = 200 ms

FIGURE 4.8: CDF of the aggregate throughputs obtained using different t-Reordering values for the SN change scenario with a dense deployment.



(D) CDF throughput for t-Reordering = 300 ms

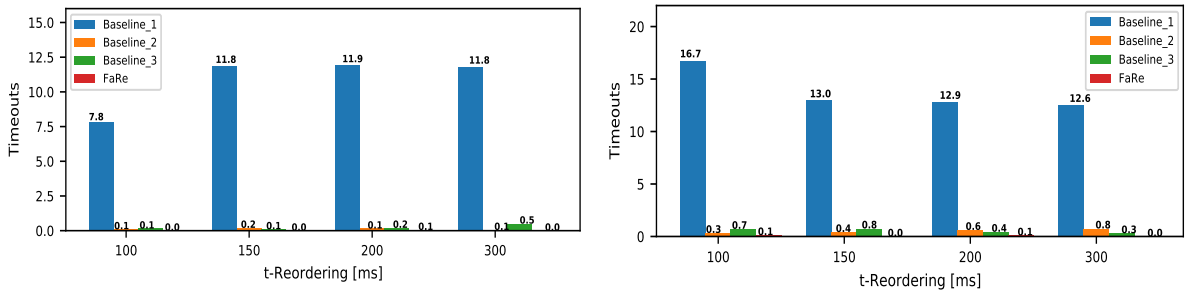
FIGURE 4.8: CDF of the aggregate throughputs obtained using different t-Reordering values for the SN change scenario with a dense deployment (cont.)

in turn harms the application's performance. In addition, even though with the other Baseline strategies, the probability of achieving zero throughput values drops to 1-2%, they still are not capable of achieving a stable throughput. Actually, abnormal peak values of more than 150 Mbps are common with the *Baseline_2* and *Baseline_3* strategies, as depicted in Fig. 4.8.

It is evident that with the *FaRe*, the aggregate throughput, regardless of the evaluated *t-Reordering* value, never goes down to zero or reaches abnormal peak values. Indeed, the *FaRe* achieves a stable throughput, which on average is around 25 Mbps for all the cases. Consequently, the application never stops receiving data even though the data splitting via the SN link is temporarily suspended.

It is worth mentioning that the periods with zero throughput and abnormal peak values, i.e., throughput larger than 45 Mbps, appear since the TCP packets received at the transport layer are not in sequence. Hence, they are buffered until TCP recovers the lost packet(s), and they can be delivered in sequence to the upper layers. In our evaluations, the lost TCP packets mainly appear when the PDCP reordering timer expires for a given PDCP PDU. Thus, data with sequence gaps, i.e., missing packets, is received at the transport layer. As shown in Fig. 4.9a, all the Baseline strategies suffer from this problem to a greater or lesser extent, being the *Baseline_1* the strategy in which more spurious PDCP PDU discards occur when the *t-Reordering* expires. Fig. 4.9a shows the average number of *t-Reordering* timeouts after running the 20 experiments for the *FaRe* and Baseline strategies.

Considering the average aggregate throughput obtained with the evaluated Baseline strategies, i.e., around 24 Mbps, the SN change events have no significant impact on the obtained result regardless of the *t-Reordering* value. However, the instantaneous throughput values significantly vary during the events mentioned above. Some applications, such as LL-eMBB or real-time applications, may not tolerate having extended periods of zero throughputs. Hence, it may be more beneficial to have a continuous data flow rather than short periods of zero and very high throughputs. Contrary to the results obtained with the Baseline strategies, the *FaRe* achieves a stable throughput with an average of 25.5 Mbps regardless of the reordering timeout value. As



(A) t-Reordering timeout declarations in the dense deployment.

(B) t-Reordering timeout declarations in the sparse deployment.

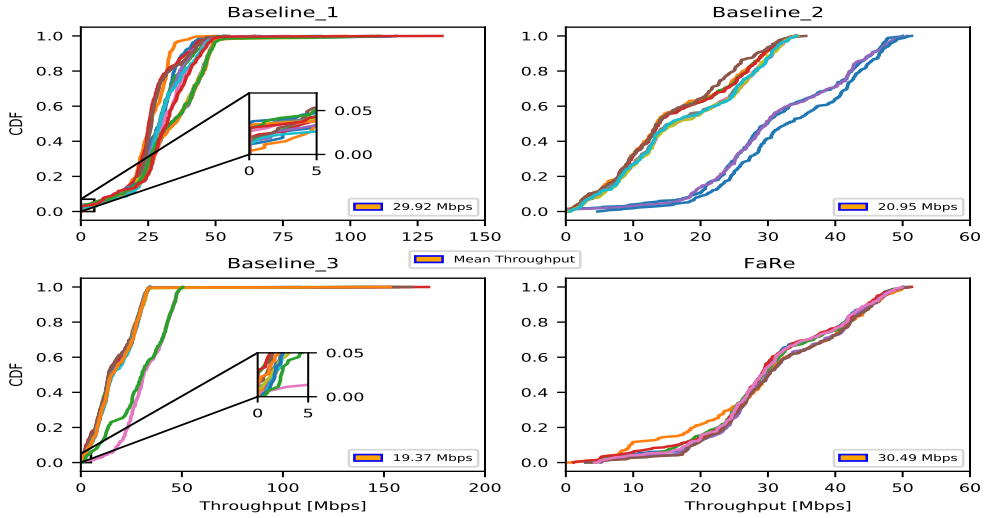
FIGURE 4.9: Average number of t-Reordering timeout declarations for the SN change scenario.

a significant advantage, the FaRe's instantaneous throughputs never reach abnormal zero and very high peak values, as illustrated in Fig. 4.8.

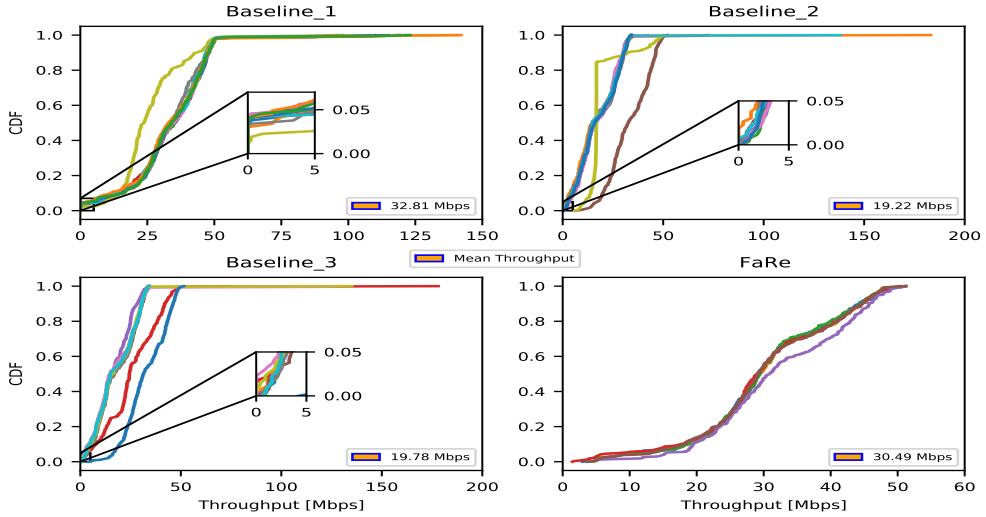
4.5.1.2 SN Change Scenario with a Sparse Deployment

As we can appreciate in Fig. 4.10, the *Baseline_2* and *Baseline_3* strategies achieve the worst performance among the benchmarked strategies regardless of the evaluated *t-Reordering* values. Indeed, their average aggregate throughput is between 18 - 20 Mbps in both cases. On the contrary, the *Baseline_1* and *FaRe* strategies achieve a higher aggregate throughput, which on average ranges from 30 to 33 Mbps for the former, and the value is around 30 Mbps for the latter. Even though the *Baseline_1* strategy achieves the best quantitative performance, it also has periods with zero and peak throughput values that reach more than 130 Mbps with all the *t-Reordering* values. Moreover, from quantitative and qualitative points of view, the *FaRe* offers the best performance regardless of the value used for the *t-Reordering*. Indeed, the *FaRe* consistently achieves an aggregate throughput around 30 Mbps without the undesirable peak throughput values. Note that with the *FaRe*, the probability of having throughput values between 20 and 40 Mbps is more than 50%.

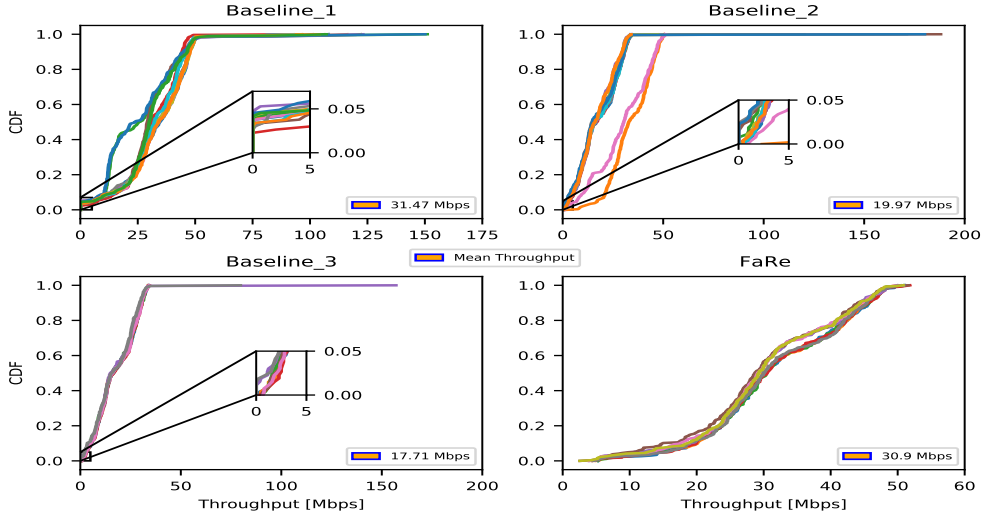
The throughput results shown in Fig. 4.10 demonstrate that all the *Baseline* strategies cannot manage continuous SN changes, especially in non-contiguous deployments where the time-to-complete the SN change procedure is large. Actually, due to the lack of coverage from the small cell, the UE will operate in SC mode until the SN change procedure is completed, causing out-of-order reception of PDCP PDUs. In this scenario, the throughput reduction or the periods with zero/peak throughput values is caused because the missing PDUs arrive at the UE after the *t-Reordering* expires, as depicted in Fig. 4.9b. As indicated in 4.9b, the reordering timeout declarations with the *Baseline_1* strategy is significantly higher than the ones obtained with the other *Baseline* strategies. This is the reason for the *Baseline_1* to achieve undesirable high peak throughput values. On the contrary, since the *FaRe* retransmits the non-received PDUs in time, they will arrive to the receiving PDCP layer before the reordering timer expires. Therefore, minimizing the out-of-order reception of PDUs and the unwanted reordering timeout declarations, as can be appreciated in Fig. 4.9b.



(A) CDF throughput for t-Reordering = 100 ms

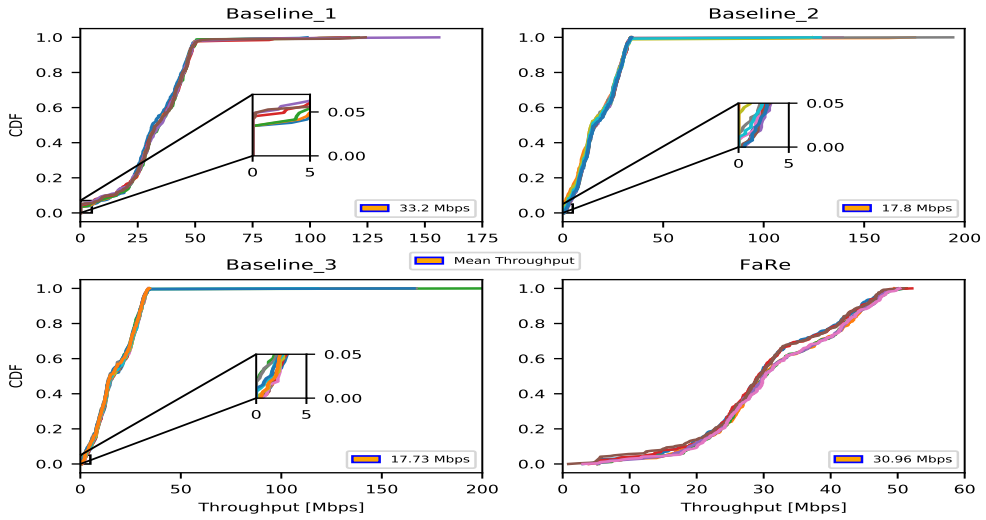


(B) CDF throughput for t-Reordering = 150 ms



(C) CDF throughput for t-Reordering = 200 ms

FIGURE 4.10: CDF of the aggregate throughputs obtained using different t-Reordering values for the SN change scenario with a sparse deployment.



(D) CDF throughput for t-Reordering = 300 ms

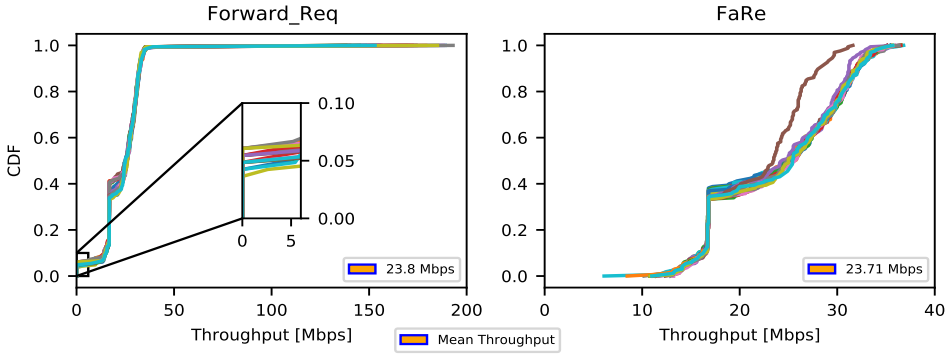
FIGURE 4.10: CDF of the aggregate throughputs obtained using different t-Reordering values for the SN change scenario with a sparse deployment (cont.)

4.5.1.3 SCG Failure Scenario with a Dense Deployment

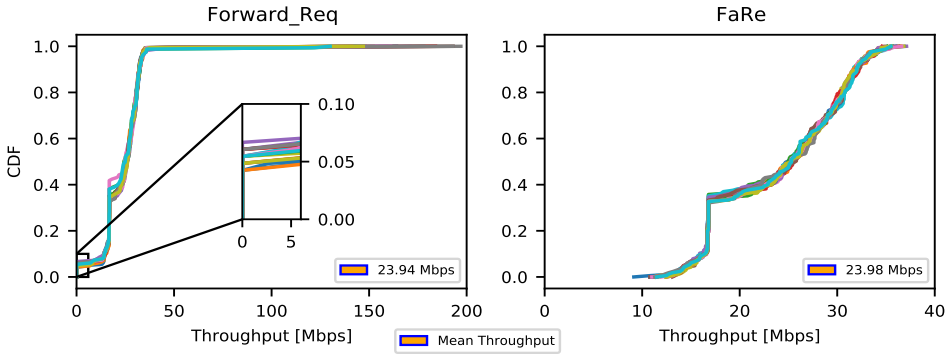
The SCG failure scenario is evaluated in a data session of 20 seconds, in which five SN change events occur before the MN receives the *SCGFailureInformation* message. As detailed in Section 4.4.1.2, the SN change events are addressed using the *Baseline_2* strategy. Thus, how the aforementioned strategies manage the SN change and failure events influence the obtained aggregate throughput. Because the UE losses connectivity with the SN after the SCG failure, the UE switches from MR-DC to SC operation, causing the aggregate throughput to drastically drop from theoretical values of 34-35 Mbps to 16-17 Mbps, as observed in Fig. 4.11.

The aggregate throughput obtained with the *FaRe* is stable and does not have abnormal zero or peak values compared to the *Forward_Req* strategy. The FaRe achieves such stability since it is aware of the last received PDCP PDU and the possible sequence gaps, the information of which is obtained from the PDCP Status Report. Hence, the MN can timely *retransmit* the non-received PDUs and fill the PDCP sequence gaps present at the UE side. Since this process is completed within a few milliseconds, the performance of *FaRe* does not depend on the *t-Reordering* value. Similar to the SN change scenario, the average throughput obtained with the *Forward_Req* and the *FaRe* are practically the same. However, some instantaneous throughput values, i.e., the zero and peak values, obtained with the *Forward_Req* strategy can seriously degrade the application's performance.

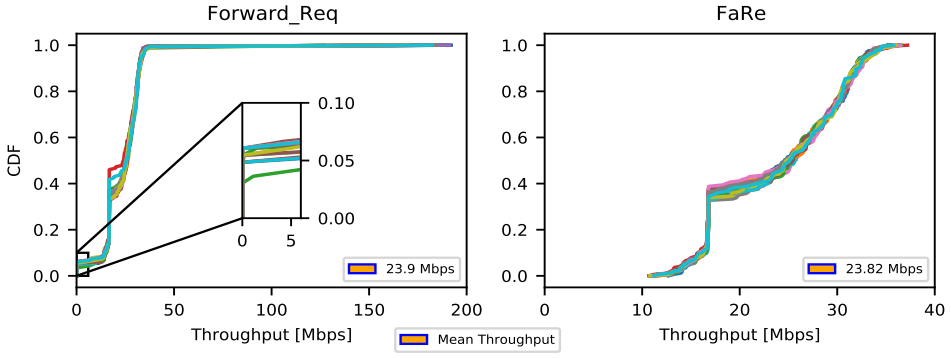
Additionally, since the UE will no longer receive data via the SN link after the UE declares an *SCG failure*, the probability of having PDCP PDU discards caused by one or more expirations of the *t-Reordering* is high. Because the *Forward_Req* strategy is not efficient in managing such an event, it has a 4-7% probability of obtaining zero throughputs, as depicted in Figs. 4.11a and 4.11d for the cases when the *t-Reordering* is configured with 100 and 300 ms, respectively. As observed in Fig.4.12a, with the *Forward_Req* strategy, the average number of timeout declarations and the subsequent effect on the discarded PDCP PDUs and buffering delay, at PDCP and Transport layers, is much larger than for the *FaRe*. The results shown in Figs. 4.11 and 4.12a reflect the inability of the *Forward_Req* strategy to correctly identify the non-received PDCP PDUs and rapidly *re-route* them to the UE via the MN's Uu. For this reason, we argue



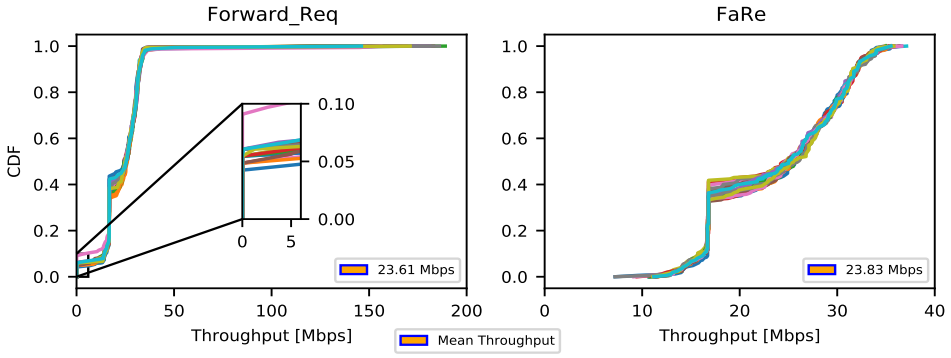
(A) CDF throughput for t-Reordering = 100 ms



(B) CDF throughput for t-Reordering = 150 ms

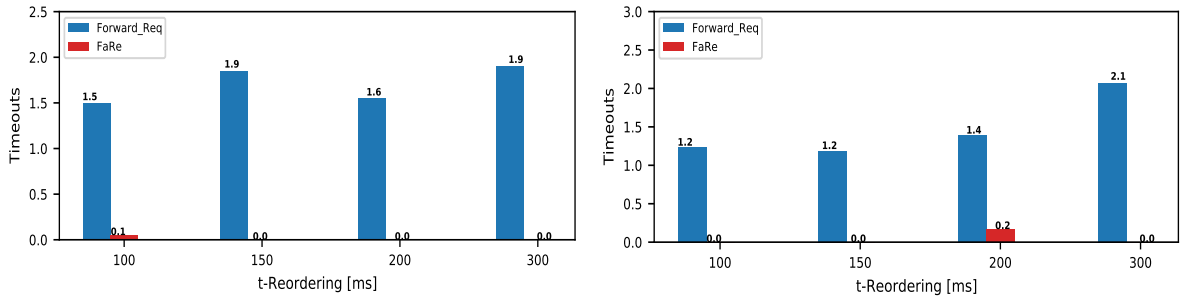


(C) CDF throughput for t-Reordering = 200 ms



(D) CDF throughput for t-Reordering = 300 ms

FIGURE 4.11: CDF of the obtained aggregate throughput using different reordering timeout values for the SCG failure case with dense deployment.



(A) t-Reordering timeout declarations in the dense deployment

(B) t-Reordering timeout declarations in the sparse deployment

FIGURE 4.12: Average number of t-Reordering timeout declarations for the SCG failure scenario.

that this method is not suitable to be used with latency- and reliability-constrained applications such as LL-eMBB applications. On the other hand, the *FaRe* has as added value the ability to efficiently serve the aforementioned applications.

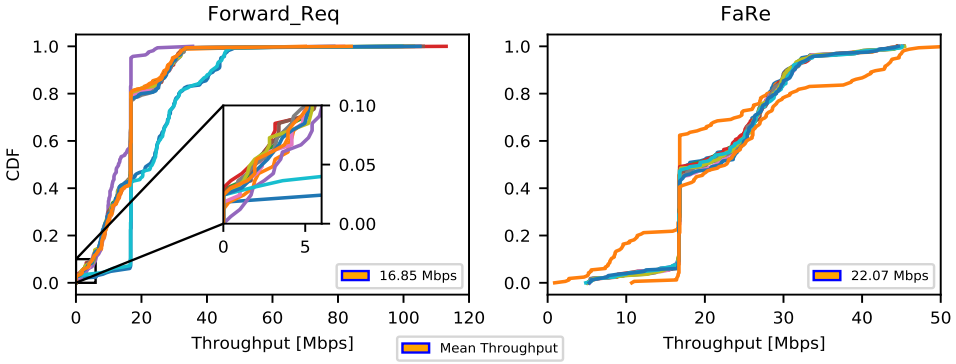
4.5.1.4 SCG Failure Scenario with a Sparse Deployment

In the sparse deployment scenario, the UE experiences seven SN change events before it declares the SCG failure and notifies with the SCGFailureInformation message to the MN. In this case, the *Forward_Req* strategy achieves an average aggregate throughput of 16-17 Mbps with all the evaluated reordering timeout values, as observed in Fig. 4.13. The low average throughput is caused mainly by the periods of zero throughput, which in turn, cause periods with very high peak values of more than 110 Mbps. Since the missing PDUs arrive at the UE after the *t-Reordering* expires, as seen in Fig. 4.9b, the PDCP layer delivers data with sequence gaps to the upper layers. These results demonstrate that the *Forward_Req* strategy is not effective managing the handovers and/or failures in the SN.

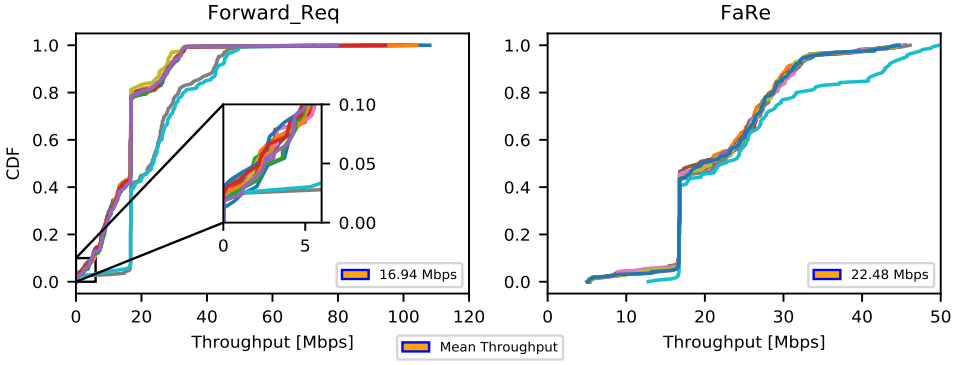
On the contrary, the *FaRe* achieves an average aggregate throughput of 21-22 Mbps, the value of which is 30% higher than the one obtained with the benchmarked strategy in all the *t-Reordering* choices. Despite the small cell coverage gaps and the SN failure, the *FaRe* does not achieve high peak throughput values or periods with zero throughputs, as shown in Fig. 4.13. Indeed, with the *FaRe*, the PDCP reordering timer does not experience any timeout declaration, as shown in Fig. 4.12b. The remarkable approach used by the *FaRe* to deal with the SCG failures makes this algorithm an effective solution to avoid the application-level performance issues.

4.5.2 Throughput Variance

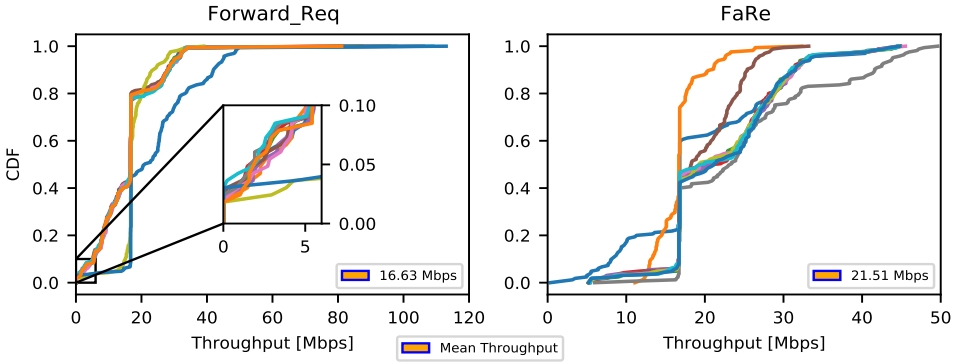
In order to study the variability of the obtained aggregate throughput during the entire data session and evince the zero and peak throughput periods, we use the R_{var}. The boxplot graphs presented in Fig. 4.14 and 4.15 correspond to the results obtained after running 20 experiments for the analyzed strategies. For this, the boxplot's median represents the average R_{var} obtained for the 20 experiments. Likewise, the first and third quartiles indicate how spread is the observed R_{var} among the experiments.



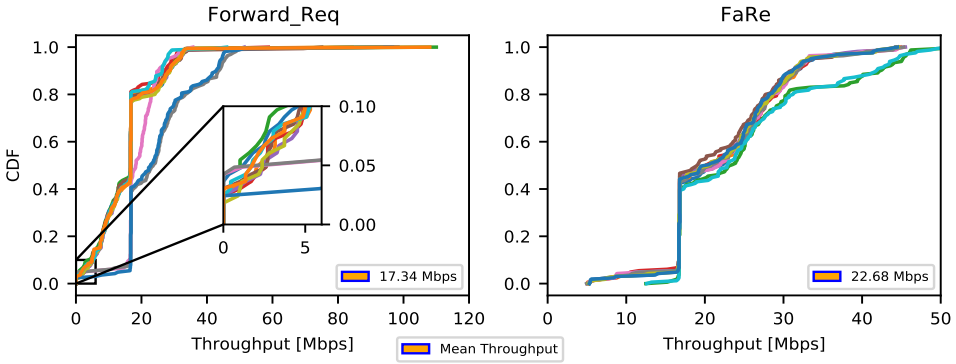
(A) CDF throughput for t-Reordering = 100 ms



(B) CDF throughput for t-Reordering = 150 ms



(C) CDF throughput for t-Reordering = 200 ms

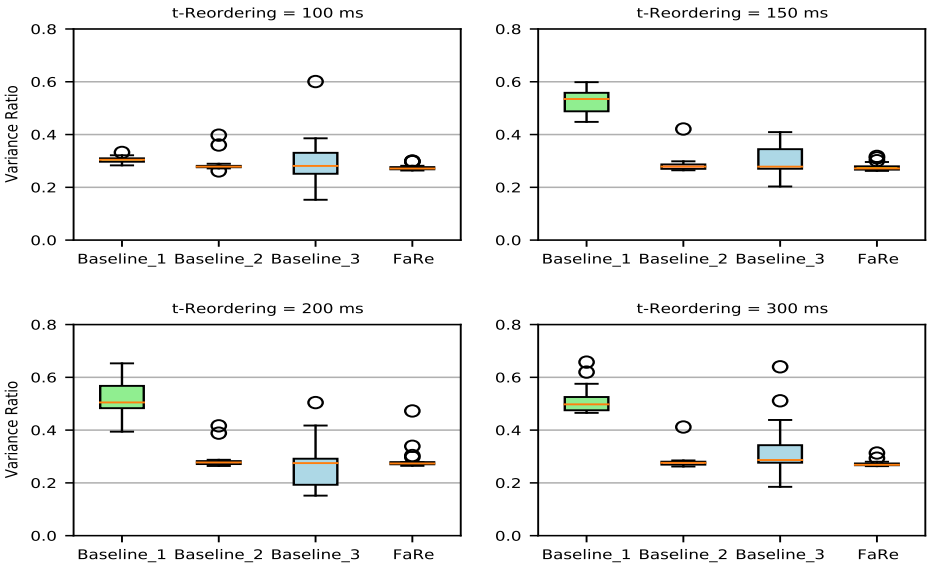


(D) CDF throughput for t-Reordering = 300 ms

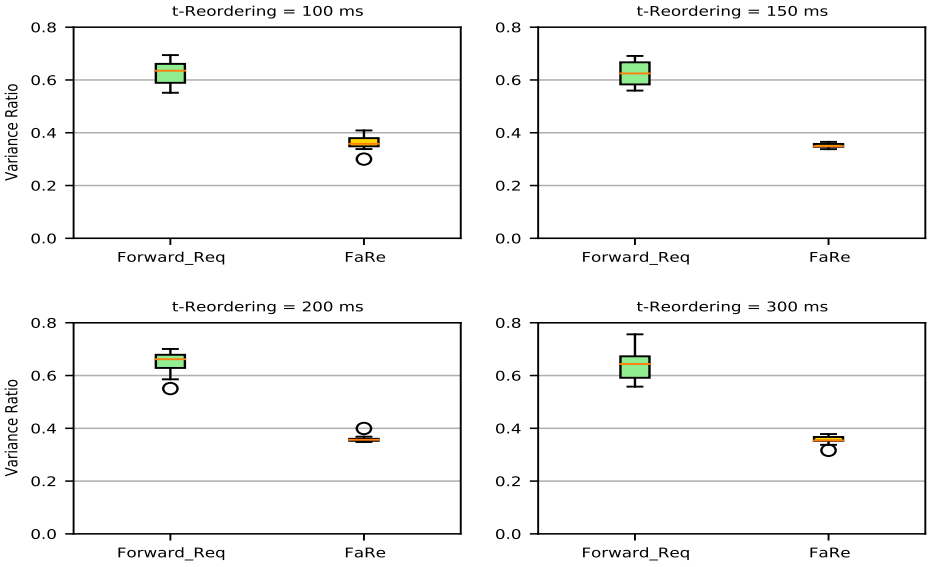
FIGURE 4.13: CDF of the obtained aggregate throughput using different reordering timeout values for the SCG failure case with sparse deployment.

4.5.2.1 Dense Deployment

We observe in Figs. 4.14a and 4.14b that the throughput obtained with the *FaRe* does not present high variability compared to the one obtained with the *Baseline* and *Forward_Req* strategies. The R_{var} for the *FaRe* is on average 0.27 for the SN change scenario and 0.35 for the SCG failure scenario, regardless of the configured *t-Reordering* value. In both cases, the individual R_{var} results are highly concentrated around the average value.



(A) Variance/Mean ratio for the *SN change* scenario



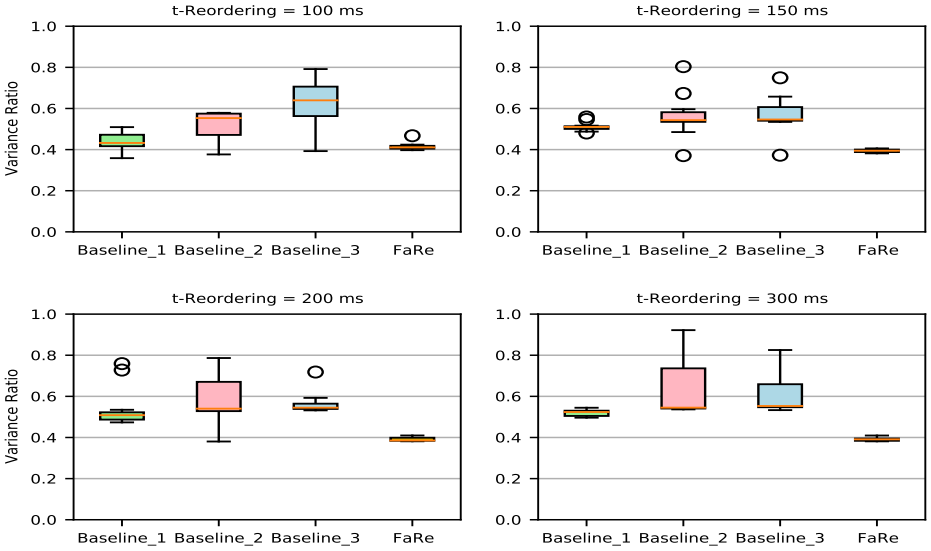
(B) Variance/Mean ratio for the *SCG failure* scenario

FIGURE 4.14: Aggregate throughput variance in the dense deployment.

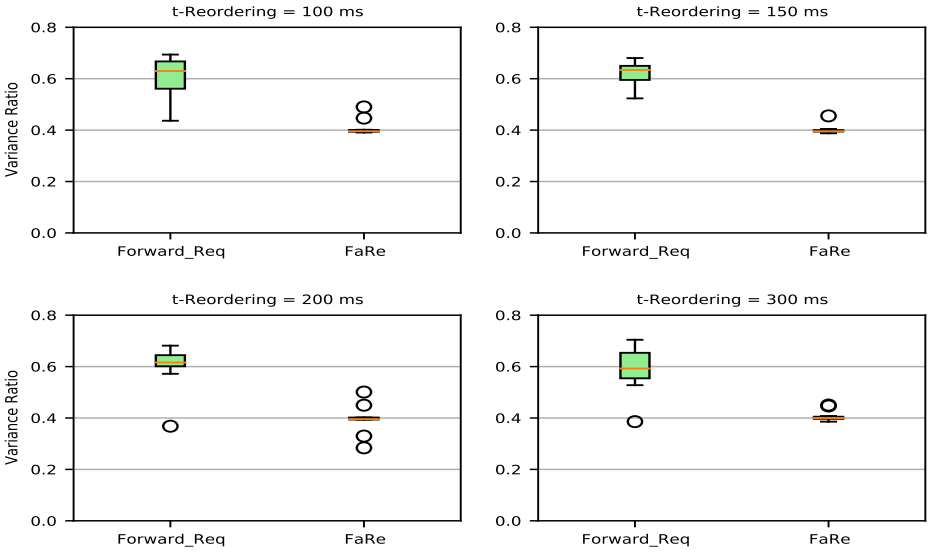
For the SN change scenario, the R_{var} obtained for the *Baseline_1* and *Baseline_3* strategies are widely spread, which means that the throughput is not stable. Moreover, according to the results shown in Fig. 4.14a, the *Baseline_2* strategy achieves a stable throughput. Nevertheless, we can see from Fig. 4.8 that such a strategy still achieves zero and peak throughputs. Furthermore,

for the SCG failure scenario, the throughput variance obtained with the *Forward_Req* strategy is on average 85% higher than with the *FaRe*, as shown in Fig. 4.14b. This instability is why the obtained throughputs are not concentrated around their average value. It is worth mentioning that the observed throughput instability for the benchmarking strategies comes from the fact that the received PDUs will spend more time in the PDCP reordering buffer before being delivered to the upper layers. The results presented in Fig. 4.14 confirm that the aggregate throughput obtained with the *FaRe* is not subject to significant and abnormal variations, which is desirable in real-time applications to satisfy a given Quality of Experience [67].

4.5.2.2 Sparse Deployment



(A) Variance/Mean ratio for the *SN change* scenario



(B) Variance/Mean ratio for the *SCG failure* scenario

FIGURE 4.15: Aggregate throughput variance in the sparse deployment.

As shown in Fig. 4.15a, the R_{var} obtained with the *Baseline_2* and *Baseline_3* strategies, respectively, indicate that the aggregate throughput in both cases is not stable, as explained in Section 4.5.1.2 and depicted in Fig. 4.10. This instability is caused by the variability of the throughput values obtained with each strategy, the values of which go from zero to more than 160 Mbps. The average R_{var} values for the *Baseline_2* and *Baseline_3* strategies are around 0.57 and 0.59, respectively, considering all the *t-Reordering* values. Similarly, the average R_{var} value for the *Baseline_1* strategy is around 0.5. Nevertheless, for the *FaRe*, the R_{var} values are concentrated around 0.4, the value of which is lower than the one obtained with the Baseline strategies.

The results obtained for the SCG failure case, depicted in Fig. 4.15b, confirm that the aggregate throughput obtained with *Forward_Req* strategy is widely spread. The average R_{var} value is around 0.6, but the single samples range from 0.4 to 0.7. In contrast, the R_{var} value obtained with the *FaRe* is highly concentrated around 0.4, except when the *t-Reordering* values is 200 ms. Similarly to the R_{var} values obtained for the dense deployment, the *FaRe* obtains a stable aggregate throughput, regardless of the reordering timeout choice, as shown in Figs. 4.13 and 4.15b.

4.5.3 Data Reliability

During data aggregation, the UE discards all the received PDCP PDUs that do not belong to the same reordering window [8], which typically happens after the *t-Reordering* expires. In this regard, if a PDCP PDU does not arrive at the UE or is discarded, the upper layers will not receive their corresponding data, which may obligate the sender to retransmit the missing data. For this reason, to study the data reliability, we focus on the reliability achieved at the UE's PDCP layer by counting the number of correctly received PDCP PDUs and comparing them with the number of PDUs that left the transmitting PDCP layer.

4.5.3.1 Dense Deployment

Tables 4.3 and 4.4 illustrate the average percentage of PDCP PDUs correctly received and its standard deviation across experiments obtained for the dense deployment after running 20 experiments with each of the evaluated strategies and reordering timeout values for the *SN change* and *SCG failure* scenarios, respectively. The results shown in Table 4.3, for the *SN change* scenario, indicate that the *Baseline_2* and *Baseline_3* strategies cannot achieve reliability of five nines for any of the reordering timeout values. At a glance, the obtained reliability seems to be suitable for applications such as LL-eMBB [64]. Nevertheless, such applications require a high and stable data rate [76], which is not possible to achieve with any of the Baseline strategies. On the contrary, the *FaRe* achieves a reliability of 100% for all reordering timeout values, except for 300 ms, with the additional benefit of having a stable and smooth data rate, as observed in Figs. 4.8 and 4.14a.

Furthermore, for the *SCG failure* scenario, the results shown in Table 4.4 demonstrate the impossibility of the *Forward_Req* strategy to offer the reliability of more than three nines, which is not the case for the *FaRe*. Indeed, with the *FaRe*, more than the 99.999% of the PDUs that left the MN's PDCP layer are correctly received at the UE, even though the SN link suddenly fails.

TABLE 4.3: PDCP Reliability for the SN change scenario in a dense deployment

Strategy	**100 ms**	**150 ms**	**200 ms**	**300 ms**
Baseline_1	99.39029	98.24219	98.20333	98.03971
δ_{B1}	0.02423	0.32628	0.24285	0.10097
Baseline_2	99.99965	99.99957	99.99950	**100**
δ_{B2}	0.00155	0.00187	0.00164	**0**
Baseline_3	99.99943	99.99965	99.99523	99.99831
δ_{B3}	0.00177	0.00154	0.01861	0.00321
FaRe	**100**	**100**	**100**	**99.99981**
δ_{FaRe}	**0**	**0**	**0**	**0.0019**

TABLE 4.4: PDCP Reliability for the SCG failure scenario in a dense deployment

Strategy	**100 ms**	**150 ms**	**200 ms**	**300 ms**
Forward_Req	99.97903	99.97978	99.97662	99.97464
δ_{Fr}	0.02771	0.02355	0.02370	0.03080
FaRe	**99.99973**	**100**	**100**	**100**
δ_{FaRe}	**0.00154**	**0**	**0**	**0**

4.5.3.2 Sparse Deployment

Tables 4.5 and 4.6 illustrate the average percentage of PDCP PDUs correctly received and its standard deviation across experiments obtained for the sparse deployment after running 20 experiments with each of the evaluated strategies and reordering timeout values for the *SN change* and *SCG failure* scenarios, respectively. Despite the highest aggregate throughput achieved by the *Baseline_1* strategy in the *SN change* scenario, as shown in Section 4.5.1.2, the obtained reliability of 98% is extremely low. Indeed, just a few applications can work in such conditions. Even though the reliability achieved by the *Baseline_2* and *Baseline_3* strategies is higher than that achieved by the *Baseline_1*, the value is not enough to support applications such as LL-eMBB. On the other hand, the *FaRe* achieves a reliability of more than 99.999% with a high and stable aggregate throughput regardless of the *t-Reordering* configuration, as illustrated in Table 4.5 and Fig. 4.10.

TABLE 4.5: PDCP Reliability for the SN change scenario in a sparse deployment

Strategy	**100 ms**	**150 ms**	**200 ms**	**300 ms**
Baseline_1	98.81777	98.82831	98.80436	98.87513
δ_{B1}	0.20580	0.26122	0.29645	0.13290
Baseline_2	99.99076	99.99151	99.97987	99.99425
δ_{B2}	0.01095	0.01004	0.02078	0.00672
Baseline_3	99.94784	99.31838	99.70093	99.51139
δ_{B3}	0.14035	1.08471	0.78479	1.12647
FaRe	**99.99968**	**100**	**99.99973**	**100**
δ_{FaRe}	**0.00082**	**0**	**0.00079**	**0**

Moreover, for the *SCG failure* case, Table 4.6 shows that the *Forward_Req* strategy achieves reliability of 99.9%, except when the *t-Reordering* value is 300 ms. In this case, such reliability

decreases to 94%. On the contrary, the *Fare* achieves a reliability of 100%, except when some random PDU losses appear, which is the case when the *t-Reordering* is configured with 200 ms.

TABLE 4.6: PDCP Reliability for the SCG failure scenario in a sparse deployment

Strategy	**100 ms**	**150 ms**	**200 ms**	**300 ms**
Forward_Req	99.97878	99.97341	99.97664	94.25316
δ_{Fr}	0.01335	0.01394	0.01354	20.59412
FaRe	**100**	**100**	**99.99835**	**100**
δ_{FaRe}	**0**	**0**	**0.00369**	**0**

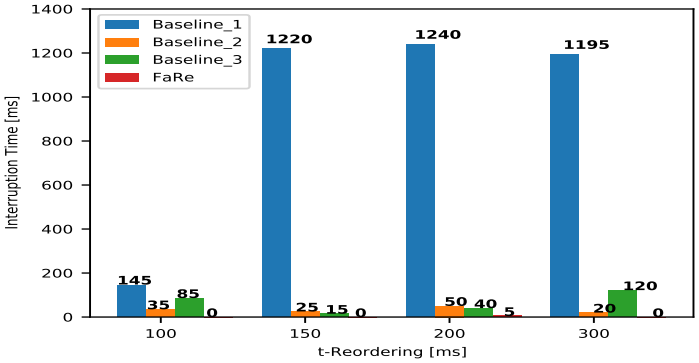
4.5.4 Data Interruption Time

To study the data interruption time, we rely on the periodic throughput reports delivered by the *iperf3* tool. In this regard, we measure the periods of zero throughputs during the entire data session for each experiment. For this, the throughput reports are periodically collected in our experiments every 100 ms for both the *SN change* and *SCG failure* scenarios. The results shown in Figs. 4.16 and 4.17 represent the average data interruption time experienced by the transport layer after running 20 experiments for the *FaRe* and every benchmarking strategy with the dense and sparse deployment, respectively.

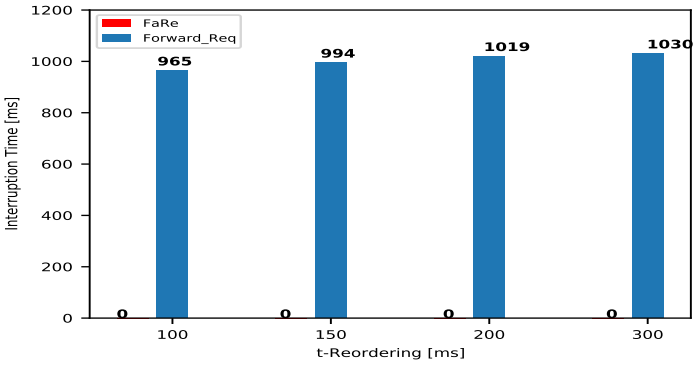
4.5.4.1 Dense Deployment

For the dense deployment scenario, the results depicted in Fig. 4.16a, for the *SN change* scenario, show the effectiveness of the *FaRe* to help the application avoid suffering from data interruption periods. On the other hand, it can be visualized that *Baseline_2* creates the lowest average interruption time among the Baseline strategies. At a glance, the obtained interruption time may not represent a significant problem in scenarios where the throughput stability and the data reliability are not the primary concern. However, for latency- and reliability-constrained applications, the *Baseline_2* is not an option to consider. Even though the Baseline strategies and the *FaRe* achieve on average a similar aggregate throughput, as shown in Fig. 4.8, the *SN change* events are not efficiently managed by the Baseline strategies. Therefore, it is going to be challenging for the applications to meet their required KPIs.

For the *SCG failure* scenario, the results obtained with the *FaRe* and illustrated in Fig. 4.16b demonstrate that it is possible to maintain a continuous data flow to the UE, even though the SN link fails. The novel capability of the *FaRe* to retransmit the non-received PDUs makes the application never stop receiving data, which on the contrary, significantly differs from the results obtained with the *Forward_Req* strategy. It is important to remark that the UE switches to SC operation after the *SCG failure* event, so the UE keeps receiving data via the MN link. Consequently, if the missing PDCP PDU(s) do not arrive at the UE in time to fill the PDCP sequence gap, the transport layer will not receive the expected data. For reliability-oriented protocols such as TCP, the transport layer will retransmit the missing data, significantly reducing the throughput and increasing the data interruption periods.



(A) Data interruption time for the SN change scenario



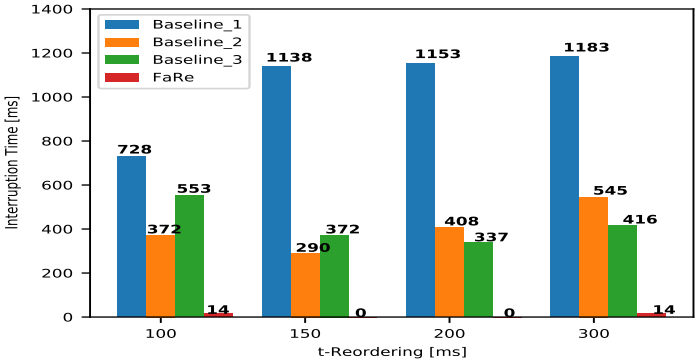
(B) Data interruption time for the SCG failure scenario

FIGURE 4.16: Data interruption time at the transport layer level with dense deployment.

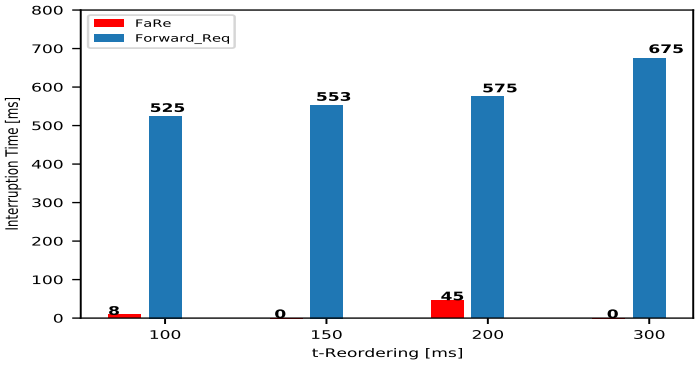
4.5.4.2 Sparse Deployment

For the sparse deployment scenario, we can appreciate in Fig. 4.17a that the *Baseline_1* strategy achieves the worse performance in terms of the periods with zero throughput values. These results confirm us again that despite achieving a high aggregate throughput, the *Baseline_1* strategy cannot offer stable results, which is not acceptable for most of the applications. The *Baseline_2* and *Baseline_3* strategies achieve similar interruption periods. However, those values are still too high to support LL-eMBB applications. On the other hand, the *FaRe* achieves 0 ms of interruption time for *t-Reordering* values of 150 and 300 ms and 14 ms for *t-Reordering* values of 100 and 300 ms. The non-zero values may result from random PDU losses that make TCP stops receiving new data. It is worth mentioning that the obtained 14 ms just represents the 3% of the lowest interruption time obtained with the *Baseline* strategies. Yet, this value does not represent an important concern for latency- and reliability-constrained applications.

For the *SCG failure* case, the results illustrated in Fig. 4.17b state that applying any data forwarding strategy when the SN fails is not effective. Instead, a timely retransmission of the missing data seems to be a significantly better approach. For this reason, the *FaRe* just suffers from very small interruption periods compared to the *Forward_Req*. The 8 and 45 ms of interruption time obtained by the *FaRe* with *t-Reordering* values of 100 and 200 ms, respectively, may be caused by a random TCP's fast retransmission action rather than any PDU lost.



(A) Data interruption time for the SN change scenario



(B) Data interruption time for the SCG failure scenario

FIGURE 4.17: Data interruption time at the transport layer level with sparse deployment.

4.5.5 The *FaRe* Implementation Impact

In Section 4.5, we have demonstrated the effectiveness of the *FaRe* to reduce the application's data interruption time during *SN change* and *SCG failure* events. The results show that it is possible to get a stable aggregate throughput with high data reliability thanks to the *FaRe*. These benefits are obtained by temporarily storing a copy of the PDCP PDUs split via the SN and timely retransmitting them via the MN when one of the events described above occurs. Since, in our experiments, the *FaRe-PDUs* are acknowledged by the MN every 5 ms, the memory requirements at the MN are not significant. In fact, the *FaRe-Buffer* usage results for the dense deployment scenario, depicted as *Buffer_Size* in Fig. 4.18, show that during a data session of 20 seconds, on average, 20 PDUs are present in the *FaRe-Buffer*. In our LTE/NR testbed, the *iperf3* tool generates packets of fixed size, creating PDCP PDUs of 1466 bytes in size. Hence, the average *FaRe-Buffer* size corresponds to 29.3 KBytes, which has a negligible impact on the MN's performance. It is worth mentioning that the buffer demand may slightly increase in case of higher throughput demands and larger bandwidth sizes.

Moreover, during an *SCG failure* event in the dense deployment, the *FaRe retransmits* via the MN's Uu, on average, 15 *FaRe-PDUs*, as can be visualized with the variable *Re-routed* in Fig. 4.18. The first of these PDUs arrive at the UE's PDCP in approximately 5-8 ms. This delay, shown as *R_Delay* in Fig. 4.18, matches with the theoretical delay, i.e., Δ^*_{FaRe}, computed with (2). Note that the measurement of this delay starts once the MN receives the *SCGFailureInformation* message and ends when the UE's PDCP layer receives the first *FaRe-PDU*.

Additionally, we noted in our experiments that in some *SN change* events, some PDUs that were initially transmitted via the SN arrive at the UE's PDCP after the *FaRe-PDUs*. This event happens when several HARQ retransmissions are required in the UE-SN path to decode the MAC transport block correctly. When this occurs, the UE can receive for the period we evaluated, on average, 6 duplicated PDUs, as depicted with the variable name *Duplicated* in Fig. 4.18. This random event has a negligible impact on the performance of the *FaRe* since the throughput, reliability, and data interruption are not affected.

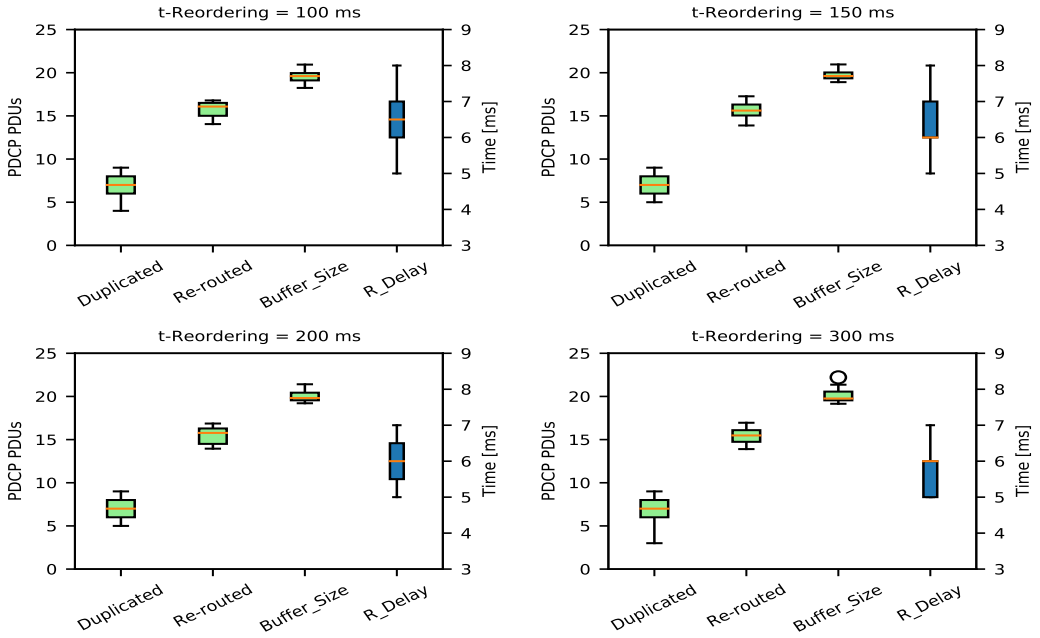


FIGURE 4.18: Different statistics for the implementation impact of the *FaRe*.

4.6 Summary

A fast data recovery mechanism that minimizes the data interruption time experienced by the application in MR-DC scenarios with mobility is presented in this chapter. The proposed mechanism, named FaRe, intelligently identifies and quickly retransmits the PDCP PDUs, at the MN, that were not received at the UE or transmitted by the SN during the aforesaid events. To accomplish this, the MN stores at its PDCP layer a copy of every PDCP PDU forwarded to the SN until an acknowledgment is received. Hence, when an SN change or SCG failure occurs, the non-delivered or non-transmitted PDCP PDUs can rapidly be transferred to the UE via the MN's Uu, thus, avoiding the time-consuming data forwarding procedures.

The experimental evaluations on an LTE/NR testbed, build using the OAI software, revealed that when the UE aggregates data, the SN change or SCG failure events affect the throughput stability and latency requirements of TCP-based applications. The results obtained in the evaluated setup showed that the proposed fast data recovery mechanism effectively reduces the harmful effects of such events for all the PDCP reordering timeout assessed values. Indeed, the *FaRe* results in zero data interruption periods for all the assessed scenarios except when the PDCP reordering timeout value is 200 ms in the SN change scenario. The efficient approach used by the *FaRe* allows the UE to achieve data reliability of more than five nines without

upper layer retransmissions and a stable and high aggregate throughput, a desirable feature to support emerging 5G applications such as low latency eMBB.

On the contrary, with the alternative strategies used to address these events, i.e., the *Baseline* and *Forward_Req* strategies, the application still stop receiving data for periods ranging from dozens to hundreds of milliseconds. These long data interruption periods make the PDCP reordering mechanism discard several PDCP PDUs, diminishing the data reliability to three nines in the *SCG failure* scenario and even to two nines in the *SN change* scenario, with the *Baseline_1* strategy. For this reason, it is challenging to meet the reliability and latency targets defined for some 5G use cases. Finally, we have demonstrated that the proposed *FaRe* mechanism effectively reduces the application's data interruption time caused by changes or failures in the SN.

Chapter 5

Conclusions and Future Work

In this chapter, we first highlight the most relevant results obtained during the PhD studies. Next, we discuss certain areas that need further research and development effort.

5.1 Conclusions

The wide range of emerging services that demand distinct QoS requirements impose challenges for the present and future mobile networks. For instance, AR/VR applications demand from the network a high average throughput, high data reliability, and relatively low packet latency. During this thesis, we have shown that Multi-connectivity technologies such as multi-radio dual connectivity can satisfy such requirements by efficiently leveraging bandwidth resources from multiple BSs simultaneously.

In Chapter 2, we conducted a comprehensive analysis of the different architecture options that enable MC operation, considering the protocol layer on which the user data can be split or duplicated. We discussed the challenges and benefits MC endures to effectively utilize the system resources from multiple BSs, which may use the same or different RAT to enhance the data rate, increase the data reliability, and provide seamless mobility. In addition, for every protocol layer, we described different 3GPP and non-3GPP MC solutions. More in detail, we went deeper into the 3GPP Multi-radio Dual Connectivity technology, where we identified that packet reordering, packet flow control, and signaling mobility management decisions are factors that can affect the application's performance when the UE aggregates data. Furthermore, to quantitatively illustrate the aforementioned challenges, we implemented a dual connectivity testbed using the LTE/NR-complaint Open Air Interface software, which is publicly available and totally modifiable. In this regard, we showed experimentally how the design decisions in terms of flow control and packet reordering impact the overall system performance when the UE aggregates data, which uses TCP and UDP transport layer protocols. Specifically, we demonstrated that the out-of-order reception of PDCP PDUs significantly degrades the obtained aggregate throughput, especially for TCP traffic.

In Chapter 3, addressing the data aggregation problem in Multi-Radio Dual Connectivity, we proposed a novel flow control mechanism that allows the transmitting PDCP layer to efficiently split the incoming user traffic via the master and secondary nodes in order to approximate the theoretical aggregate throughput. To make the novel proposal agnostic to the MR-DC architecture option, we use the RLC buffering delay and MAC SDU size statistics from the

RLC and MAC layers, respectively, of both BSs for the traffic splitting decisions. By doing this, the proposed mechanism does not define any new signaling messages between the UE and BSs, making it 3GPP-compliant. One of the great advantages of the proposed flow control mechanism is to maintain an upper limit on the RLC buffering delay created at both BSs. In consequence, the delay difference between two PDCP PDUs with consecutive sequence numbers can be easily controlled. This feature has a tremendous impact on the performance of the PDCP reordering mechanism since the aggregate throughput becomes independent of the PDCP reordering timeout value. Indeed, it also avoids the excessive buffering delay created at the receiver PDCP layer by the out-of-order reception of PDCP PDUs. The aforesaid performance was experimentally analyzed as realistically as possible using a real radio link channel quality indicator dataset extracted from a drive-test tool operating in a real deployment scenario.

In Chapter 4, we analyzed the challenges mobility events and radio link failures bring to MR-DC to meet the KPIs targets defined for different services. In this regard, we figured out that when the UE aggregates data, handover and signal blockage events that occur at the secondary node can make the UE temporarily operate in SC mode until the SN change or RRC connection reestablishment procedures are completed. Since these procedures are time-consuming, the out-of-order reception of PDCP PDUs increases, making the application stop receiving data for up to hundreds of milliseconds. In addition, we observed, to the best of our knowledge, that no studies have considered the impact of the aforesaid events on the performance of the upper layers during MR-DC operation. Indeed, the interruption time caused by those events has been addressed only from the physical layer perspective. To bridge the gap on this topic, we presented a novel fast data recovery mechanism, which makes the MN capable of timely retransmitting, to the UE, the missing PDCP PDUs that may arise during handovers and/or failures in the secondary node. Our experimental evaluations using an LTE/NR testbed under 3GPP-defined scenarios and link channel quality variations, based on traces extracted from a Nokia-proprietary system-level-simulator, revealed that our proposed mechanism can offer a stable aggregate throughput with near-zero data interruption time at the application level and data reliability of at least five nines without upper layers retransmissions. Therefore, the proposed fast data retransmission mechanism makes it possible to improve the latency and reliability requirements during mobility events. In this regard, mobile network operators can take advantage of the proposed mechanism to support emerging 5G applications, e.g., low latency eMBB, and to avoid losing data during SCG activation/deactivation procedures.

Lastly, MR-DC promises significant improvements to the overall system performance, but MNOs need to devise efficient solutions.

5.2 Future Research Directions

In this thesis, we have endeavored to find efficient solutions that allow the UE in MR-DC operation effectively aggregate data from multiple BSs. We have evaluated our solutions using a real LTE/NR testbed under realistic conditions in 3GPP-defined scenarios. Nonetheless, some questions remain open and need further research efforts. In our opinion, the most important aspects can be summarized as follows:

- The dense and heterogeneous deployments expected for future networks open the possibility for the UE to simultaneously connect to more than two 3GPP and non-3GPP BSs in order to achieve higher data rates or improve data reliability. Unfortunately, implementing such multiple connections requires additional hardware, e.g., more than two 5G

radio interfaces, and user plane integration at the UE that can significantly increase the device complexity, thus, its final cost. Hence, resulting non-commercially attractive for the MNOs. Instead of implementing more radio interfaces at the UE, techniques that allow the UE to switch the connection between BSs, e.g., to a new SN, can be further explored as solutions.

- The Capacity and Congestion Aware flow control mechanism, proposed in Chapter 3, demands continuous up-to-date statistics from the MAC and RLC layers of both BSs to effectively adapt the splitting decisions to the variable radio link conditions. Implementing data analytics and machine learning techniques can help the flow control mechanism to make better decisions using less frequent signaling message exchange.

- To support a high aggregate throughput without sacrificing from the latency requirements seems significantly challenging in MR-DC operation. In this thesis, we have only focused on maximizing the obtained aggregate throughput. Therefore, future research efforts can focus on finding flow control solutions that allow the MN to split the incoming traffic to maximize the throughput while minimizing the latency.

- In Chapter 4, we proposed a solution to reduce the data interruption time experienced at the application level when SN changes/failures occur. If similar events occur at the MN, the UE will stop receiving data from both BSs, degrading the application's performance. Hence, it is imperative to study solutions to support the Inter-Master Node HOs and MCG failures.

- The SCG activation/deactivation technique is a promising solution that helps the UE to reduce energy consumption. However, if the flow control mechanism, e.g., at the MN, is unaware of such actions, it can keep splitting PDCP PDUs via the SN, the PDUs of which might not arrive at the UE. This lack of cross-layer communication between the transmitting MAC and PDCP layers can cause out-of-order reception of PDCP PDUs, which, in turn, can increase the application's data interruption time.

- The 3GPP states that the master node must control all the management aspects required to support MR-DC operation. This centralized management approach limits the flexibility that the BS acting as secondary node has to add, modify, or change the SN functionalities into a new BS. Even critical aspects, such as failure in the SN-UE link, i.e., SCG failure, cannot be independently managed by the secondary node. Allowing the SN to perform some of the tasks mentioned above can alleviate the processing requirements at the MN, reduce the impact of signaling message exchange on the backhaul, and minimize the delay required to complete such procedures. Therefore, we consider that exploring new ideas on this topic can improve the efficiency of MR-DC operation.

- The flow control and fast data recovery mechanisms proposed in Chapters 3 and 4, respectively, have been widely evaluated in 3GPP-defined scenarios and under realistic conditions. However, we have used the *iperf3* tool to generate saturated TCP and UDP traffic. Hence, it remains open to evaluating the performance of those mechanisms using real traffic, in which the communication link also experiences packet losses, delays, etc.

Bibliography

[1] ITU-R. IMT Vision – Framework and overall objectives of the future development of IMT for 2020 and beyond. *ITU-R M.2083-0*, 0:https://www.itu.int, 2015.

[2] Accenture Strategy. The Impact of 5G on the European Economy. Technical report, Accenture Strategy, 2021.

[3] IHS Markit. The role of 5G in a post-pandemic world economy. Technical report, IHS Markit, 2020.

[4] Claudio Rosa, Klaus Pedersen, Hua Wang, Per Henrik Michaelsen, Simone Barbera, Esa Malkamäki, Tero Henttonen, and Benoist Sébire. Dual connectivity for LTE small cell evolution: Functionality and performance aspects. *IEEE Communications Magazine*, 54 (6):137–143, 2016. ISSN 01636804. doi: 10.1109/MCOM.2016.7498101.

[5] 3GPP. TS 36.300 - V16.7.0 - LTE; Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2. Technical report, 3GPP, 2022.

[6] 3GPP. TS 37.340 - V15.7.0 - Universal Mobile Telecommunications System (UMTS); LTE; 5G; NR; Multi-connectivity; Overall description; Stage-2, 2019.

[7] Osman N.C. Yilmaz, Oumer Teyeb, and Antonino Orsino. Overview of LTE-NR Dual Connectivity. *IEEE Communications Magazine*, 57(6):138–144, 2019. ISSN 15581896. doi: 10.1109/MCOM.2019.1800431.

[8] 3GPP. TS 38.323 - V15.3.0 - 5G; NR; Packet Data Convergence Protocol (PDCP) specification (Release 15), 2018.

[9] Carlos Pupiales, Daniela Laselva, and Ilker Demirkol. Capacity and Congestion Aware Flow Control Mechanism for Efficient Traffic Aggregation in Multi-Radio Dual Connectivity. *IEEE Access*, 9:114929–114944, 2021. ISSN 21693536. doi: 10.1109/ACCESS.2021.3105177.

[10] Ravilla Dilli. Analysis of 5G Wireless Systems in FR1 and FR2 Frequency Bands. *2nd International Conference on Innovative Mechanisms for Industry Applications, ICIMIA 2020 - Conference Proceedings*, pages 767–772, mar 2020. doi: 10.1109/ICIMIA48430.2020.9074973.

[11] Carlos Pupiales, Daniela Laselva, Quentin De Coninck, Akshay Jain, and Ilker Demirkol. Multi-Connectivity in Mobile Networks: Challenges and Benefits. *IEEE Communications Magazine*, 59(11):116–122, nov 2021. ISSN 15581896. doi: 10.1109/MCOM.111.2100049.

[12] Carlos Pupiales, Daniela Laselva, and Ilker Demirkol. Fast Data Recovery for Improved Mobility Support in Multi-Radio Dual Connectivity. *IEEE Access*, pages 1–1, 2022. ISSN 2169-3536. doi: 10.1109/ACCESS.2022.3204027.

[13] Carlos Pupiales, Walter Nitzold, Clemens Felber, and Ilker Demirkol. Software-Based Implementation of Dual Connectivity for LTE. In *Proceedings - 2019 IEEE 16th International Conference on Mobile Ad Hoc and Smart Systems Workshops, MASSW 2019*, pages 178–179. Institute of Electrical and Electronics Engineers Inc., nov 2019. ISBN 9781728141213. doi: 10.1109/MASSW.2019.00048.

[14] Carlos Pupiales and Ilker Demirkol. Efficient Traffic Aggregation for Dual Connectivity, 2021.

[15] 3GPP. TS 38.420 - V17.0.0 - 5G; NG-RAN; Xn general aspects and principles. Technical report, 3GPP, 2022.

[16] 3GPP. TS 36.425 - V15.0.0 - Universal Mobile Telecommunications System (UMTS); LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); X2 interface user plane protocol, 2018.

[17] Small Cell Forum. Small Cell Forum 7.0 17:25. Technical Report January, Small Cell Forum, 2016.

[18] Jin Liu, Kelvin Au, Amine Maaref, Jun Luo, Hadi Baligh, Hui Tong, Alexander Chassaigne, and Javier Lorca. Initial Access, Mobility, and User-Centric Multi-Beam Operation in 5G New Radio. *IEEE Commun. Mag.*, 56(3):35–41, 2018. ISSN 01636804. doi: 10.1109/MCOM.2018.1700827.

[19] Thorsten Biermann, Luca Scalia, Changsoon Choi, Holger Karl, and Wolfgang Kellerer. CoMP clustering and backhaul limitations in cooperative cellular mobile access networks. *Pervasive and Mobile Computing*, 8:662–681, 2012. doi: 10.1016/j.pmcj.2012.03.005.

[20] Mohammed Hirzallah, Marwan Krunz, Balkan Kecicioglu, and Belal Hamzeh. 5G New Radio Unlicensed: Challenges and Evaluation. *IEEE Transactions on Cognitive Communications and Networking*, 2020. ISSN 23327731. doi: 10.1109/TCCN.2020.3041851.

[21] Marie Theres Suer, Christoph Thein, Hugues Tchouankem, and Lars Wolf. Multi-Connectivity as an Enabler for Reliable Low Latency Communications - An Overview. *IEEE Communications Surveys and Tutorials*, 22(1):156–169, 2020. ISSN 1553877X. doi: 10.1109/COMST.2019.2949750.

[22] 3GPP. TS 23.501 - V16.6.0 - 5G; System architecture for the 5G System (5GS). Technical report, 3GPP, 2020.

[23] Alan Ford, Costin Raiciu, Mark Handley, and Olivier Bonaventure. TCP Extensions for Multipath Operation with Multiple Addresses. Technical report, IETF, 2013.

[24] Quentin De Coninck and Olivier Bonaventure. Multipath QUIC: Design and evaluation. In *CoNEXT 2017 - Proceedings of the 2017 13th International Conference on emerging Networking EXperiments and Technologies*, volume 7, pages 160–166, 2017. ISBN 9781450354226. doi: 10.1145/3143361.3143370.

[25] Adnan Aijaz. Packet Duplication in Dual Connectivity Enabled 5G Wireless Networks: Overview and Challenges. *IEEE Communications Standards Magazine*, 3(3):20–28, sep 2019. ISSN 24712825. doi: 10.1109/MCOMSTD.001.1700065.

[26] 3GPP. TS 38.425 - V16.2.0 - 5G; NG-RAN; NR user plane protocol, 2020.

[27] Mikel Irazabal, Elena Lopez-Aguilera, Ilker Demirkol, Robert Schmidt, and Navid Nikaein. Preventing RLC Buffer Sojourn Delays in 5G. *IEEE Access*, 9, 2021. ISSN 21693536. doi: 10.1109/ACCESS.2021.3063769.

[28] 3GPP. TR 38.913; 5G; Study on scenarios and requirements for next generation access technologies, 2020.

[29] Lucas Chavarria Gimenez, Per Henrik Michaelsen, Klaus I. Pedersen, Troels E. Kolding, and Huan Cong Nguyen. Towards Zero Data Interruption Time with Enhanced Synchronous Handover. *IEEE Vehicular Technology Conference*, 2017-June, 2017. ISSN 15502252. doi: 10.1109/VTCSpring.2017.8108504.

[30] 3GPP. TR 36.881-V14.0.0; Study on latency reduction techniques for LTE. Technical report, 3GPP, 2016.

[31] 3GPP. TS 38.331 - V15.5.1 - 5G; NR; Radio Resource Control (RRC); Protocol specification, 2019.

[32] Ericsson. R2-170271. Technical report, Ericsson, Spokane, 2017.

[33] Ericsson. Tackling fast recovery from radio link failure - Ericsson, 2020. URL https://www.ericsson.com/en/blog/2020/9/fast-recovery-from-radio-link-failure.

[34] Navid Nikaein, Mahesh K Marina, Saravana Manickam, Alex Dawson, Raymond Knopp, and Christian Bonnet. OpenAirInterface: A Flexible Platform for 5G Research. *ACM SIGCOMM Computer Communication Review*, pages 33–38, 2014.

[35] Rutgers University. Open-Access Research Testbed for Next-Generation Wireless Networks (ORBIT), 2022. URL https://www.orbit-lab.org/.

[36] Darijo Raca, Jason J. Quinlan, Ahmed H. Zahran, and Cormac J. Sreenan. Beyond throughput: A 4G LTE Dataset with Channel and Context Metrics. In *Proceedings of the 9th ACM Multimedia Systems Conference, MMSys 2018*, pages 460–465, New York, NY, USA, jun 2018. Association for Computing Machinery, Inc. ISBN 9781450351928. doi: 10.1145/3204949.3208123.

[37] Asif Ali Laghari, Hui He, and Muhammad Ibrahim Channa. Measuring Effect of Packet Reordering on Quality of Experience (QoE) in Video Streaming. *3D Research*, 9(3), 2018. ISSN 20926731. doi: 10.1007/s13319-018-0179-6.

[38] Intel. 3GPP Contribution: R2-132859 Throughput evaluation and comparison of with and without UP bearer split. Technical Report August, Intel Corporation, 2013.

[39] Hua Wang, Claudio Rosa, and Klaus I. Pedersen. Dual connectivity for LTE-advanced heterogeneous networks. *Wireless Networks*, 22(4):1315–1328, 2016. ISSN 15728196. doi: 10.1007/s11276-015-1037-6.

[40] Meng Shiuan Pan, Tzu Ming Lin, Chun Yuan Chiu, and Ching Yen Wang. Downlink Traffic Scheduling for LTE-A Small Cell Networks With Dual Connectivity Enhancement. *IEEE Communications Letters*, 20(4):796–799, 2016. ISSN 10897798. doi: 10.1109/LCOMM.2016.2522404.

[41] Kien Nguyen, Mirza Golam Kibria, Jing Hui, Kentaro Ishizu, and Fumihide Kojima. Minimum Latency and Optimal Traffic Partition in 5G Small Cell Networks. *IEEE Vehicular Technology Conference*, 2018-June:1–5, 2018. ISSN 15502252. doi: 10.1109/VTCSpring.2018.8417644.

[42] Roberto P. Antonioli, Emanuel B. Rodrigues, Diego A. Sousa, Igor M. Guerreiro, Carlos F.M. e Silva, and Francisco R.P. Cavalcanti. Adaptive bearer split control for 5G multi-RAT scenarios with dual connectivity. *Computer Networks*, 161:183–196, 2019. ISSN 13891286. doi: 10.1016/j.comnet.2019.07.005.

[43] Roberto P Antonioli, Igor M Guerreiro, Diego Aguiar Sousa, Rodrigues Emanuel Bezerra, Carlos F. M. e Silva, Tarcísio Ferreira Maciel, and Francisco Rodrigo Porto Cavalcanti. User-assisted bearer split control for dual connectivity in multi-RAT 5G networks. *Wireless Networks*, 26, 2020. doi: 10.1007/s11276-020-02283-6.

[44] Yixuan Wang, Changyin Sun, Fan Jiang, and Jing Jiang. Blocking- And Delay-aware Flow Control Using Markov Decision Process. *2020 IEEE/CIC International Conference on Communications in China, ICCC 2020*, pages 905–910, aug 2020. doi: 10.1109/ICCC49849.2020.9238831.

[45] Boram Jin, Segi Kim, Donggyu Yun, Hojin Lee, Wooseong Kim, and Yung Yi. Aggregating LTE and Wi-Fi: Toward Intra-Cell Fairness and High TCP Performance. *IEEE Transactions on Wireless Communications*, 16(10):6295–6308, oct 2017. ISSN 15361276. doi: 10.1109/TWC.2017.2721935.

[46] David Lopez-Perez, Daniela Laselva, Eugen Wallmeier, Paivi Purovesi, Petteri Lunden, Elena Virtej, Piotr Lechowicz, Esa Malkamaki, and Ming Ding. Long Term Evolution-Wireless Local Area Network Aggregation Flow Control. *IEEE Access*, 4:9860–9869, 2016. ISSN 21693536. doi: 10.1109/ACCESS.2016.2643690.

[47] 3GPP. TS 38.214 - V16.5.0 - 5G; NR; Physical layer procedures for data, 2021.

[48] Moustafa M Nasralla, Nabeel Khan, and Maria G Martini. Content-aware downlink scheduling for LTE wireless systems: A survey and performance comparison of key approaches. *Computer Communications*, 130:78–100, 2018. doi: 10.1016/j.comcom.2018.08.009.

[49] Sassan Ahmadi. *LTE-Advance: A Practical Systems Approach 3GPP LTE Releases 10 and 11 Radio Access Technologies LTE-Advanced.* Elsevier, first edition, 2014. ISBN 9780124051621.

[50] 3GPP. TS 38.321 - V15.7.0 - NR; Medium Access Control (MAC) protocol specification, 2019.

[51] A. Misra, T. Ott, and J. Baras. Effect of exponential averaging on the variability of a RED queue. *IEEE International Conference on Communications*, 6:1817–1823, 2001. ISSN 05361486. doi: 10.1109/icc.2001.937105.

[52] Leonard Kleinrock. *Queuing Systems, Volume I: Theory.* John Wiley, 1975.

[53] Leonard Kleinrock. Internet congestion control using the power metric: Keep the pipe just full, but no fuller. *Ad Hoc Networks*, 80:142–157, 2018. ISSN 15708705. doi: 10.1016/j.adhoc.2018.05.015.

[54] 3GPP. TS 38.322 - V15.5.0 - NR; Radio Link Control (RLC) Protocol Specification, 2019.

[55] 3GPP. TS 36.213 - V15.11.0 - LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures, 2020.

[56] 3GPP. TS 36.322 - V15.1.0 - LTE; Radio Link Control (RLC) Protocol specification, 2018.

[57] K. Salah and F. Haidari. On the performance of a simple packet rate estimator. In *AICCSA 08 - 6th IEEE/ACS International Conference on Computer Systems and Applications*, pages 392–395, 2008. ISBN 9781424419685. doi: 10.1109/AICCSA.2008.4493563.

[58] Dominik Kaspar. *Multipath aggregation of heterogeneous access networks.* PhD thesis, University of Oslo, 2012.

[59] Stephen Hemminger. Network Emulation with NetEm. In *Proceedings of Australia's 6th National Linux Conference*, Canberra, 2005.

[60] 3GPP. TR 36.932 - V12.1.0 - LTE; Scenarios and requirements for small cell enhancements for E-UTRA and E-UTRAN, 2014.

[61] Florian Kaltenberger. OpenAirInterface 5G Overview, Installation, Usage. In *OpenAirInterface Workshop*, Beijing, 2019.

[62] D. F. Williamson, R. A. Parker, and J. S. Kendrick. The box plot: A simple visual method to interpret data. *Annals of Internal Medicine*, 110(11):916–921, 1989. ISSN 00034819. doi: 10.7326/0003-4819-110-11-916.

[63] ETRI. R2-1903906. Technical report, 3GPP, Xi'an, 2019.

[64] MediaTek. RWS-210094. Technical report, MediaTek, 2021.

[65] Menglei Zhang, Michele Polese, Marco Mezzavilla, Jing Zhu, Sundeep Rangan, Shivendra Panwar, and Michele Zorzi. Will TCP Work in mmWave 5G Cellular Networks? *IEEE Communications Magazine*, 57(1):65–71, jan 2019. ISSN 15581896. doi: 10.1109/MCOM.2018.1701370.

[66] 3GPP. TR 26.918 - V16.0.0 - Universal Mobile Telecommunications System (UMTS); LTE; Virtual Reality (VR) media services over 3GPP. Technical report, 3GPP, 2020.

[67] Michele Polese, Marco Giordani, Marco Mezzavilla, Sundeep Rangan, and Michele Zorzi. Improved Handover Through Dual Connectivity in 5G mmWave Mobile Networks. *IEEE Journal on Selected Areas in Communications*, 35(9):2069–2084, sep 2017. ISSN 07338716. doi: 10.1109/JSAC.2017.2720338.

[68] Ping Jung Hsieh, Wei Shih Lin, Kuang Hsun Lin, and Hung Yu Wei. Dual-Connectivity Prevenient Handover Scheme in Control/User-Plane Split Networks. *IEEE Transactions on Vehicular Technology*, 67(4):3545–3560, apr 2018. ISSN 00189545. doi: 10.1109/TVT.2017.2778065.

[69] Tariq Mumtaz, Shahabuddin Muhammad, Muhammad Imran Aslam, and Nazeeruddin Mohammad. Dual Connectivity-Based Mobility Management and Data Split Mechanism in 4G/5G Cellular Networks. *IEEE Access*, 8:86495–86509, 2020. ISSN 21693536. doi: 10.1109/ACCESS.2020.2992805.

[70] Rasha El Banna, Hussein M. Elattar, and Mohamed Aboul-Dahab. Handover Scheme for 5G Communications on High Speed Trains. *2020 5th International Conference on Fog and Mobile Edge Computing, FMEC 2020*, pages 143–149, apr 2020. doi: 10.1109/FMEC49853.2020.9144880.

[71] Kaiqiang Qi, Tingting Liu, Chenyang Yang, Shiqiang Suo, and Yuanfang Huang. Dual connectivity-aided proactive handover and resource reservation for mobile users. *IEEE Access*, 9:36100–36113, 2021. ISSN 21693536. doi: 10.1109/ACCESS.2021.3062196.

[72] Mikhail Gerasimenko, Dmitri Moltchanov, Margarita Gapeyenko, Sergey Andreev, and Yevgeni Koucheryavy. Capacity of Multiconnectivity mmWave Systems with Dynamic Blockage and Directional Antennas. *IEEE Transactions on Vehicular Technology*, 68(4): 3534–3549, apr 2019. ISSN 19399359. doi: 10.1109/TVT.2019.2896565.

[73] Vitaly Petrov, Dmitrii Solomitckii, Andrey Samuylov, Maria A. Lema, Margarita Gapeyenko, Dmitri Moltchanov, Sergey Andreev, Valeriy Naumov, Konstantin Samouylov, Mischa Dohler, and Yevgeni Koucheryavy. Dynamic Multi-Connectivity Performance in Ultra-Dense Urban mmWave Deployments. *IEEE Journal on Selected Areas in Communications*, 35(9):2038–2055, sep 2017. ISSN 07338716. doi: 10.1109/JSAC.2017.2720482.

[74] Mustafa F. Ozkoc, Athanasios Koutsaftis, Rajeev Kumar, Pei Liu, and Shivendra S. Panwar. The Impact of Multi-Connectivity and Handover Constraints on Millimeter Wave and Terahertz Cellular Networks. *IEEE Journal on Selected Areas in Communications*, 39(6): 1833–1853, jun 2021. ISSN 15580008. doi: 10.1109/JSAC.2021.3071852.

[75] Siyoung Choi, Jin Ghoo Choi, and Saewoong Bahk. Mobility-aware analysis of millimeter wave communication systems with blockages. *IEEE Transactions on Vehicular Technology*, 69(6):5901–5912, jun 2020. ISSN 19399359. doi: 10.1109/TVT.2020.2987082.

[76] Sunmi Jun, Yong Seok Choi, and Heesang Chung. Considerations on Ultra broadband, High reliable and Low latency services in 6G system. In *International Conference on ICT Convergence*, volume 2021-Octob, pages 1552–1554. IEEE Computer Society, 2021. ISBN 9781665423830. doi: 10.1109/ICTC52510.2021.9621081.

[77] ETRI. R2-1903906 MIT in eMBB Handover. Technical report, 3GPP, Xi'an, 2019.

[78] Docomo. RWS-210234. Technical report, Docomo, 2021.

[79] Samsung. RWS-210183. Technical report, Samsung, 2021.

[80] 3GPP. TR 38.802-V14.2.0; Study on New Radio Access Technology Physical Layer Aspects. Technical Report Release 14, 3GPP, 2017.

[81] 3GPP. TR 36.872 - V1.0.0; Small cell enhancements for E-UTRA and E-UTRAN - Physical layer aspects. Technical Report Release 12, 3GPP, 2013.