IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 30, 2021, accepted August 8, 2021, date of publication August 16, 2021, date of current version August 24, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3105177

Capacity and Congestion Aware Flow Control Mechanism for Efficient Traffic Aggregation in Multi-Radio Dual Connectivity

CARLOS PUPIALES[1,2], DANIELA LASELVA[3], AND ILKER DEMIRKOL[4], (Senior Member, IEEE)

[1]Department of Network Engineering, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain
[2]FICA, Universidad Técnica del Norte, Ibarra 100105, Ecuador
[3]Nokia, 9220 Aalborg, Denmark
[4]Department of Mining, Industrial and ICT Engineering, Universitat Politècnica de Catalunya, 08240 Manresa, Spain

Corresponding author: Carlos Pupiales (carlos.pupiales@upc.edu)

This work was supported in part by the Secretaría de Educación Superior, Ciencia, Tecnología e Innovación (SENESCYT), Ecuador.

ABSTRACT Multi-Radio Dual Connectivity (MR-DC) is a crucial 3GPP technology that enables traffic aggregation to leverage the radio resources of two base stations (BSs), thereby increasing the per-user data rate. However, the traffic aggregation management in MR-DC is left up to vendor implementation. In this paper, we show that enabling an efficient traffic aggregation method is crucial to increase the throughput performance of both TCP- and UDP-based applications in MR-DC operation. Targeting the state-of-the-art gap on this topic, we propose a flow control mechanism, which efficiently aggregates traffic based on the assigned radio resources and buffering delay statistics of both BSs. The proposed traffic aggregation mechanism is applicable irrespective of the employed MR-DC architecture option, MAC packet scheduler design, and transport layer protocol in use. By means of exhaustive testbed experiments, we show that the proposed method achieves at least 85% and 95% of the theoretical aggregate throughput when employing MR-DC for TCP and UDP traffic, respectively.

INDEX TERMS Flow control, multi-connectivity, dual-connectivity, 5G, traffic aggregation.

I. INTRODUCTION

Achieving high data rates is one of the requirements for 5G networks, especially to serve the enhanced mobile broadband (eMBB) use cases. Indeed, the 3rd Generation Partnership Project (3GPP) targets a minimum peak data rate of 20 Gbps for a single user in the downlink [1]. In addition, 5G networks should accommodate a large number of users with such peak data rates in a given geographical area. However, this poses significant challenges due to the limited radio resources. Different methods to increase the per-user data rate can be considered such as increasing the bandwidth, utilizing cell densification, and improving the spectral efficiency [2]. However, the scarcity of the spectrum resources, deployment costs, and hardware complexities are factors that constrain the implementation of the above-indicated approaches. For instance, carrier aggregation can increase the user data rate, but it requires higher bandwidth resources. Likewise, beamforming and massive multiple-input multiple-output (mMIMO) systems can improve the signal-to-interference-plus-noise ratio (SINR) and thus, can increase the obtained data rate. However, high manufacturing costs and hardware complexities still limit their wide adoption.

Dual Connectivity (DC) [3] and Multi-Radio Dual Connectivity (MR-DC) [4] are 3GPP solutions, which can increase the per-user data rate without the need for additional bandwidth resources acquisition or significant hardware complexities. In both technologies, the user equipment (UE) can simultaneously transfer user plane (UP) data with two BSs. Hence, two data streams, each from a different BS, can be combined into a single data stream (a.k.a traffic aggregation) in order to enhance the data rate of the UE. To accomplish this goal, one of the two BSs acts as an anchor entity that manages the control plane (CP) related aspects and transfers the UP traffic of the UE using only one BS or both BSs. In DC, both BSs employ the same 3GPP radio access technology (RAT), i.e., Long Term Evolution (LTE) or 5G New Radio (NR), while MR-DC also considers a heterogeneous RAT setup. For simplicity, we use MR-DC to refer to both DC and MR-DC since the working principle in both technologies is the same.

The associate editor coordinating the review of this manuscript and approving it for publication was Ting Wang.

In MR-DC, the anchor BS is called the master node (MN), and the assisting BS is called the secondary node (SN).

The 3GPP has defined distinct UP architecture options to enable MR-DC operation in different use cases and deployment scenarios. However, we noted that traffic aggregation is only possible using the so-called split data radio bearer (DRB) configuration [2], [5], where packet data units (PDUs) can be transferred through both BSs at the same time. Note that the DRB is the logical connection used to transport UP traffic between the BS and the UE. In this regard, the Packet Data Convergence Protocol (PDCP) layer is responsible for the data splitting in the transmitting entity, and the data aggregation in the receiving entity [3], [4]. This PDCP layer serves as a common entity for two independent protocol stacks formed by the radio link control (RLC), medium access control (MAC), and physical (PHY) layers [2]. In this work, we call to the *lower layer protocol stack*, i.e., RLC, MAC, and PHY, as *communication path*.

Aggregating PDCP PDUs using MR-DC may result in a throughput improvement. Indeed, under ideal conditions, the obtained throughput, i.e., the ideal aggregate throughput, would be equal to the sum of the throughputs achieved at each BS when employing single connectivity (SC) operation. However, due to signal interruptions or traffic fluctuations, the temporarily varying conditions in the communication paths can cause under-utilized links on one side and high RLC buffering delays on the other side, both of which negatively affect the obtained throughput.

Additionally, using a non-ideal backhaul (BH) connection between the two BSs, i.e., a backhaul with limited capacity and non-zero delay, is an important challenge to achieve the ideal aggregate throughput. In fact, the delay added by the backhaul can cause the PDUs traversing the MN to experience different sojourn times compared to the PDUs traversing the SN. This delay difference, in turn, may result in out-of-order packet reception, which would degrade the performance of reliability-oriented transport layer protocols such as the Transport Control Protocol (TCP). Indeed, because of out-of-order issues, the aggregate throughput can even be lower than the one achieved in SC, as we showed in [6]. To minimize this problem, the receiving PDCP layer should reorder the packets before delivering them to the upper layers. Unlike TCP, the User Datagram Protocol (UDP) throughput is not affected by this problem. However, the out-of-order deliveries can be treated as packet loss by the application, and thus they would affect the perceived quality of delay-sensitive applications using UDP [7].

Because of the out-of-order packet deliveries and the temporarily varying radio conditions, the incoming user traffic should be distributed through both communication paths using a flow control method working at the MN's PDCP layer. Hence, the PDCP layer can smartly and dynamically split the traffic according to the instantaneous assigned radio resources at each BS. Since the MN's PDCP layer is not aware of the variability of the radio link conditions, especially the ones from the SN, the flow control decisions become challenging. For instance, splitting the incoming PDUs based on a Round Robin logic can fairly utilize the resources in both BSs. However, this logic may be helpful if the radio link conditions in both communication paths and the instantaneous data rates are approximately the same between the UE and both BSs, which reduces its applicability.

Most of the relevant research studies in MR-DC propose flow control methods that do not consider traffic aggregation's upper layer throughput performance. 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 not realistic. Therefore, the impact of such proposals on the performance of transport layer protocols, e.g., for TCP, has not been appropriately addressed. Nevertheless, such MR-DC traffic aggregation studies are necessary, since most of the Internet traffic worldwide is based on TCP [8]. Moreover, the assumptions and simplifications that state-of-the-art solutions use 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 network protocol configurations. Actually, 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 the mentioned gap in the literature, in this paper, we propose a flow control mechanism for traffic aggregation in MR-DC, which is aware of the instantaneous available radio resources and the RLC buffering delay to efficiently and dynamically split the UP traffic via the MN and SN. Our Capacity and Congestion Aware (CCW) flow control mechanism is agnostic to the MR-DC architecture option, MAC packet scheduler design, and transport layer protocol in use. Additionally, our CCW flow control does not define any new signaling feedback from the UE for the traffic splitting decisions. We evaluate the throughput performance of CCW under different channel bandwidth assignment scenarios for both TCP- and UDP-based traffic in a mobile network testbed. We utilize a radio link channel trace [9], demonstrating the adaptability of the CCW to the dynamic radio link conditions and dynamic assignment of radio resources. Through testbed experiments, we show that the CCW achieves more than the 85% and 95% of the ideal aggregate throughput for TCP- and UDP-based traffic, respectively, regardless of the evaluated scenario. Even though the evaluations were conducted only in a single RAT scenario, i.e., LTE-DC, our CCW flow control can also be used in a multi-RAT scenario.

In summary, the contributions of this paper are manifold:

- We analyze the challenges to effectively aggregate traffic in MR-DC, considering transport and application layer aspects.
- We propose a flow control algorithm for traffic aggregation in MR-DC that addresses the identified challenges.
- We evaluate the proposed CCW flow control mechanism using the LTE/5G-NR compliant Open Air Interface

software and commodity hardware. The performance of the CCW is evaluated against state-of-the-art and benchmark solutions. These analyses are conducted using a real radio link Channel Quality Indicator (CQI) dataset.

The rest of this paper is structured as follows. Section II presents the description of MR-DC from a protocol layer perspective along with the challenges it faces to achieve the ideal aggregate throughput. Additionally, this section also analyzes the related work in the field. In Section III, the design principles of the CCW are detailed. Furthermore, the evaluation framework and the testbed setup details are discussed in Section IV. Moreover, the experiments conducted and their results are shown in Section V. Finally, the conclusions of the work are presented in Section VI.

II. RELATED WORK

This section describes the main technological aspects and the challenges that an efficient MR-DC operation demands for traffic aggregation along with the state-of-the-art solutions proposed for such purpose.

A. MR-DC IN 5G NR

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 core networks (CN), i.e., Evolved Packet Core (EPC) for LTE and 5G Core (5GC) for NR. In the latter, heterogeneous BSs are connected to either the EPC or 5GC. Different MR-DC architecture options are defined by 3GPP depending on which CN and anchor BS is used. Fig. 1 depicts the currently supported MR-DC architecture options, where LTE-DC and NR-DC options are considered as single RAT, and E-UTRA-NR Dual Connectivity (EN-DC), NR-E-UTRA Dual Connectivity (NE-DC), and Next Generation-RAN E-UTRA-NR Dual Connectivity (NGEN-DC) are considered as multi-RAT DC solutions [4].

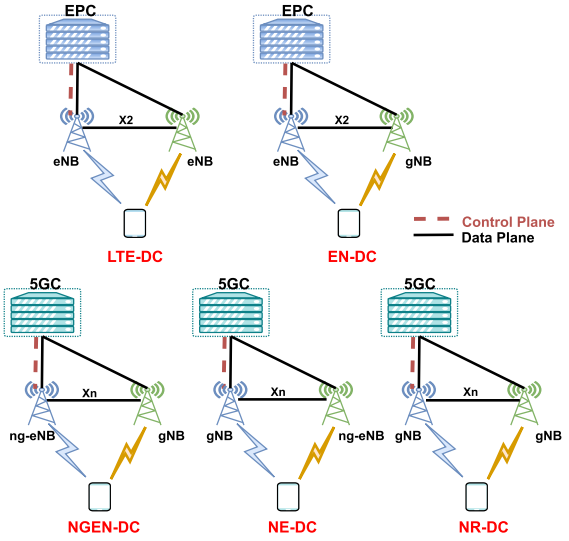


FIGURE 1. MR-DC architecture options.

1) CONTROL PLANE ASPECTS

In any MR-DC architecture option, the UE is connected to the CN via the MN, while it establishes a Radio Resource Control (RRC) connection with the MN and SN. 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 Radio Resource Management (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 [2].

2) USER PLANE ASPECTS

From the CN's perspective, the UP traffic is transferred to/from the Radio Access Network (RAN) using either the MN or SN. On the other hand, from the UE's perspective, the UP data can go through either via one BS or both BSs simultaneously. This depends on the configuration, which can be on a per DRB level and on the dynamic traffic aggregation decision [4]. In this regard, the UE can use radio bearers that belong only to MN or SN, or to both BSs. For the latter case, 3GPP specifies the split DRB, which allows the UE to consume radio resources from both the MN and SN at the same time [4]. 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). In this work, we consider the first case, which allows aggregating traffic using the radio resources from both BSs simultaneously.

3) MR-DC CHALLENGES

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

a: PACKET REORDERING AND DISCARDING

The communication paths associated with different BSs experience different delays due to the heterogeneous radio link conditions, the difference in the assigned radio resources, and the procedures that each BS may employ to tackle the data transmission, which may depend on the used RAT. This delay difference can cause out-of-order arrival of PDUs, which may have a negative impact on the performance of transport layer protocols such as TCP. Therefore, PDUs should be reordered at the receiving side before being delivered to the upper layers.

To address the out-of-order delivery, 3GPP has specified a PDCP-level reordering mechanism to use in MR-DC operation [10]. For this, a reordering timer and window are used at the UE to wait for the missing PDU(s) whenever a PDU sequence number gap in the reordering window is detected. We note that if the missing PDUs are not received while the reordering timer is running, a reordering timeout is declared. Hence, the PDUs present in the reordering window are delivered to the upper layers. 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, 3GPP does not specify neither a particular timeout value nor a procedure to choose one. Actually, an inadequate timeout value choice can degrade the obtained aggregate throughput. For instance, a large timeout value can lead to an excessive waiting time for a delayed PDU, which causes bufferbloat. On the other hand, a small timeout value can cause spurious PDU discards.

b: DYNAMIC FLOW CONTROL

In MR-DC, the PDCP flow control mechanism is in charge of determining the amount of PDCP PDUs to transmit via each communication path. An inadequate flow control mechanism may cause out-of-order deliveries or under-utilized links, and thus, negatively impacting the overall system performance. In this sense, the flow control method should dynamically split the user traffic according to the assigned radio resources and radio link conditions at the MN and SN. To achieve the ideal aggregate throughput, the flow control method should maintain the transmitting RLC buffers with sufficient data to prevent under-utilized links while avoiding congestion that can increase the PDUs' sojourn times [11]. Likewise, as much as possible, the flow control should minimize the PDU reordering needs at the receiver.

B. 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 [12], the traffic is split using a fixed ratio that does not consider the dynamism of the wireless link. The BS with the largest capacity is configured to transmit 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 [13], 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, the SN maintains its buffer with enough data to be scheduled at each transmission opportunity. Hence, the SN sends the data requests to MN based on a trade-off between the buffering time and the possibility of link starvation.

Moreover, the authors in [14] propose a downlink traffic scheduling method to maximize the network throughput and keep a fair throughput distribution among the UEs connected to the SN. For this, 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 attempts to minimize the end-to-end delay instead of enhancing the perceived data rate is proposed in [15]. 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 [16] present a utility-based algorithm that splits the user traffic in order to maximize user satisfaction. For this, utility weights based on certain Quality of Service (QoS) levels are computed. Although this work provides useful insights, the proposed algorithm does not specify the correlation used to calculate the splitting ratio between the utility weights and the QoS metrics. Furthermore, an improved version of the solution presented by [16] is introduced in [17]. 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 [16]. Hence, it also faces the same limitations already indicated. Moreover, a flow control solution that intends to reduce the blocking state in a mmWave link is proposed in [18]. This algorithm aims to offload the user traffic from one BS to another when the mmWave link is unavailable. The results shows 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 LTE WLAN Aggregation (LWA), the insights from which can be used for MR-DC. For instance, in [19], authors propose a solution that pursues intra-cell throughput fairness for TCP-based traffic. To achieve that, the algorithm uses the UE's feedback to estimate the delay, and then, PDUs go 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 in comparison with 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 [20]. 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 paths. The evaluations are conducted only for TCP traffic in a co-located scenario, i.e., where the LTE and WLAN RATs are located in the same BS. Hence, it is

not known what is the impact of the backhaul delay on the performance of this algorithm would be.

Our literature survey shows the need for a new flow control algorithm that would target achieving the ideal aggregate throughput for TCP and UDP traffic. Hence, we propose a new flow control algorithm that efficiently aggregates UP traffic regardless of the MR-DC architecture option, MAC packet scheduler design, transport layer protocol in use as shown through testbed experiments.

III. CCW FLOW CONTROL ALGORITHM

A. TARGETED CHALLENGES

Traffic aggregation in MR-DC entails the aggregation of PDCP PDUs, which are transported via the MN and SN, and belong to the same DRB, i.e., split DRB. Therefore, an ideal aggregate throughput would be equal to the sum of the throughputs achieved at the MN and SN, respectively, when employing SC operation. This obtained throughput must be, at least, higher than the highest SC throughput regardless of the transport layer protocol and application in use; otherwise, it would be beneficial to use SC instead. Since the UE is connected to two different BSs, the RRM procedures such as packet scheduling and link adaptation are independently managed at each BS according to the experienced radio link conditions and load level at each Transmission Time Interval (TTI) or slot. As a consequence, the PDUs traversing the two BSs will experience different sojourn times depending on the communication path's delay. The difference between the communication paths' delays will cause the PDUs to arrive out of order at the receiving PDCP layer, for which a reordering mechanism will be needed to ensure in-sequence packet delivery to the upper layers. Hence, the higher is the delay difference between the communication paths; the higher is the percentage of out-of-order PDUs.

The delay difference is influenced by the backhaul latency and the packet sojourn delay in each BS. The former, in the typical deployments, is expected to introduce few milliseconds of one-way latency. Whereas the latter depends on the RLC buffer occupancy. Indeed, the sojourn delay can significantly vary due to the fluctuation of radio link conditions and user contention for radio resources. Indeed, these fluctuations cause only part of the pending data to be transferred to/from the UE can be accommodated at each TTI. Thus the remaining data is buffered in the RLC buffer until the next transmission opportunity. Suppose the channel capacity is low compared to the previously assigned rate and assuming that the incoming data keeps piling up at the same speed. In that case, the amount of the buffered data may temporarily increase. This effect may lead to congest the link that suffers from radio link condition deterioration and may exacerbate the delay difference between communication paths. In this sense, the backhaul latency and the congestion caused by the buffering delay in the RLC buffers are significant aspects to consider for splitting the incoming PDCP PDUs via both BSs.

To maximize the aggregate throughput, both the MN and SN should maintain a continuous data flow with the UE. In light of that, the flow control mechanism should feed the corresponding RLC buffers with a sufficient number of PDUs so that they can be transmitted at each TTI, but without increasing the load level, i.e., the congestion, in such buffers. In this regard, our proposed flow control mechanism will address the challenges mentioned above for MR-DC.

B. CCW PRINCIPLES DESIGN

Our CCW flow control mechanism aims to aggregate traffic from the MN and SN and to 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 uplink. The CCW flow control mechanism 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 (2) 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 is designed to operate in BSs that use the same TTI value, e.g., 1 ms. However, it could be extended to operate with any arbitrary TTI/slot duration. The main building blocks of the CCW are described in the following.

1) CAPACITY AND CONGESTION ESTIMATION

In both LTE and NR, 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 then 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 [21], [22]. Since the radio link conditions and assigned PRBs may be continuously changing, the TBS changes as well. In consequence, in poorer radio link conditions and/or higher cell loads, fewer data can be transmitted towards the UE. For this reason, unsent PDUs should be buffered 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 [23].

Typically, the MAC packet scheduler shares the available PRBs among the active UEs and among the radio bearers configured for the UE [24], 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 Service Data Unit (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 [25], [26]. Note 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) [27] 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:

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

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 [28]. Therefore, the RLC buffering delay (D_q) of the split DRB is defined at each TTI n according to:

$$D_q[n] = \frac{RLC_{BS}[n]}{SDU_{DC}[n]}, \tag{2}$$

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:

$$D^*_q[n] = \alpha \times D_q[n] + (1-\alpha) \times D^*_q[n-1], \tag{3}$$

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. 2 illustrates the representation of SDU_{DC} and its EWMA-based averages, i.e., SDU^*_{DC}.

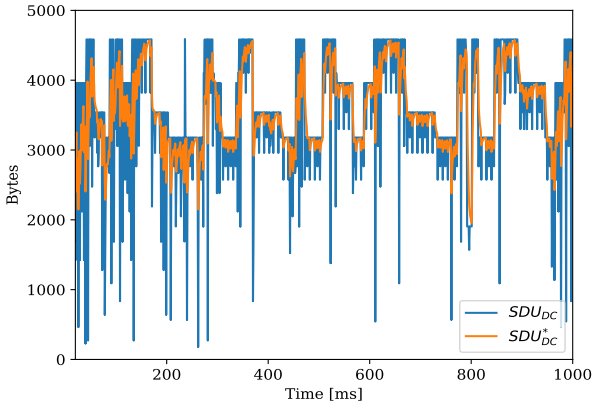


FIGURE 2. Estimation of SDU^*_{DC}.

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 can be transmitted through the $Xn/X2$ interface using the downlink data delivery status (DDDS) procedure, which the 3GPP has specified for flow control report purposes in [29].

Since the SN and MN are 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 sense, to avoid further delay to such a report, it should be transmitted every TTI. However, this significantly increases the backhaul traffic and its capacity requirement. Since the availability of up-to-date CQI values depends on the CSI reporting type configured for the UE, i.e., periodic or aperiodic, 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) TRAFFIC SPLITTING

The time-varying traffic characteristics, e.g., packet arrival rates and packet 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 a 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 PDCP, where the incoming PDCP PDUs are temporarily stored in a FIFO manner before they are split. Hence, it is possible to periodically split the PDUs via both communication paths according to the MAC SDU size and RLC buffering delay statistics 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 [30].

For this, the CCW takes UP traffic splitting decisions periodically, where at the beginning of that period, the CCW defines the amount of data to send to each RLC layer. In this paper, this period is named as traffic splitting time interval (T_{CCW}). In this sense, 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 RLC buffering delay is equal to $2 \times BH + TTI$. Hence, having a large T_{CCW} value may be inefficient in order to adapt to the radio link condition changes. Since the CC_R message arrives every Δt_{cc}, the splitting decisions should be taken every $T_{CCW} = \Delta t_{cc}$ as well.

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 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.

- 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).

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

- 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. The statistical information required by the CCW per split DRB is available at the 4G/5G BSs since a BS can handle multiple UEs configured with multiple DRBs, where each DRB has its own PDCP and RLC instances [10], [31].

IV. EVALUATION FRAMEWORK

In order to evaluate and validate the proposed CCW flow control algorithm for MR-DC, we implement the UP functionalities of the LTE-DC architecture option with split DRB using the LTE/NR compliant Open Air Interface Software (OAI) for eNB, UE, and EPC [32]. We experimentally evaluate this implementation using the ORBIT Testbed facilities [33]. The details of the software implementation, testbed setup, and evaluation scenarios are exposed in this section.

A. SOFTWARE IMPLEMENTATION OF DC

We extended the existing protocol stack implementation of OAI for the eNB and UE in order to support the UP

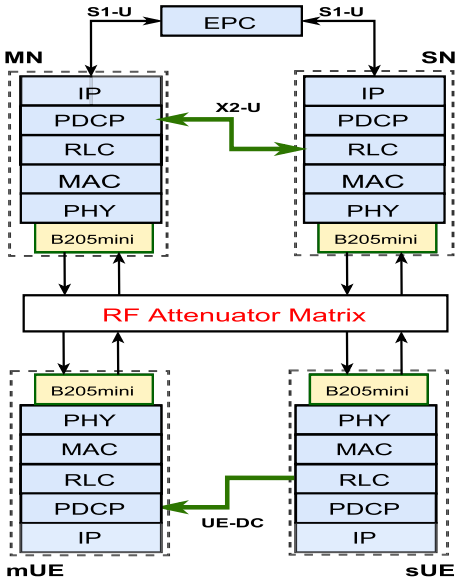


FIGURE 3. Testbed architecture.

functionalities of LTE-DC with split DRB for the downlink traffic. For this, we implemented in the PDCP layer of the UE the 3GPP reordering mechanism described in [10]. Moreover, we incorporated new functionalities in the eNB software to support the operation of distinct flow control algorithms. The code used in the eNB, UE, and EPC for the experiments can be found at *https://github.com/Carlitops/DC_Flow_Control*. The LTE-DC architecture used in this implementation is depicted in Fig. 3 and described as follows.

1) SOFTWARE IMPLEMENTATION FOR THE eNB

At the eNB side, we implement the X2-U network interface to handle the transfer of PDCP PDUs from the eNB acting as MN to the one acting as SN. For this, we use UDP encapsulation instead of the GPRS Tunnelling Protocol. Note that this simplification does not affect the obtained experimental results given the negligible protocol header overhead difference. In this context, 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 the capacity and congestion report 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 forwards it to the PDCP layer for further processing. In this setup, the X2-U latency is equal to the BH latency, and the X2-U link capacity is approximately 1 Gbps.

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., for the RLC, MAC, and PHY protocol stacks. 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. 3, 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 work for such purpose. The UE-DC interface works in a similar way as the X2-U interface does, i.e., it transports PDUs from the sUE to the mUE using a UDP encapsulation. The delay added by the communication link between sUE and mUE hosts in our testbed is approximately 0.15 ms one-way, which is a negligible delay in our evaluations.

B. TESTBED SETUP

Five hosts from the ORBIT Testbed [33] are used to represent the UEs, eNBs, and EPC, and are connected using a Gigabit-Ethernet switch. 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 eNB and UE hosts. Each SDR is electromagnetically isolated from the others, but they are connected using a programmable RF attenuator matrix, model JFW 50PMA-012 [33]. This setup allows having an isolated single input single output (SISO) RF path between each eNB and UE, which leads to a negligible inter-cell interference. Further configurations and parameters used in this study for the BSs are illustrated in Table 1.

TABLE 1. Basic configuration for the eNBs.

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	10 MHz
Max number of HARQ transmissions	4 [25]
RLC mode	UM

C. EXPERIMENTAL SETUP

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 the SN according to a Round Robin (*RR*) approach. The PDCP sequence numbers are used for the splitting decision making, i.e., the PDUs with even sequence numbers are transferred via the MN and the ones 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 [20]. This *Delay-based* algorithm relies on the UE status reports from both 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, 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 (2), to represent the delay used in [20]. This delay is measured and forwarded every 5 ms to the MN's PDCP layer, the periodicity of which matches the *time between UE status reports* (Δt) used in [20]. Note that there may be details and parameters to tune for the Delay-based flow control approach that cannot be determined from [20]. Nevertheless, the design of the delay-based flow control approach is largely implemented and configured according to the methodology and values described in [20], which are listed as follows: $\Delta t = 5$ ms, maximum queuing delay $d_{max} = 30$ ms, and fairness $\beta = 0$.

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 [27], [34].

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}) [8] 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 efficiency. In our setup, these values represent the average throughputs obtained over different runs.

The definition of A_{ben} for the MR-DC case, according to [8], is detailed in (4).

$$A_{ben}(FC) = \begin{cases} \dfrac{T_{DC} - T_{SC}^{max}}{T_{SC}^{min}} & \text{if } T_{DC} \geq T_{SC}^{max} \\ \dfrac{T_{DC} - T_{SC}^{max}}{T_{SC}^{max}} & \text{if } T_{DC} < T_{SC}^{max} \end{cases} \tag{4}$$

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.

D. EVALUATION SCENARIOS

For the performance evaluations, we define two main scenarios. The first scenario, *scenario A*, considers that both BSs have the same available bandwidth for the UE, i.e., 10 MHz. In the second scenario, *scenario B*, BSs have different available bandwidths, where 5 MHz and 10 MHz are used for the MN and SN, respectively.

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 in [9]. 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 eNB 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 eNBs 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. 4 illustrates the CQI values reported to the MN and SN in one experiment.

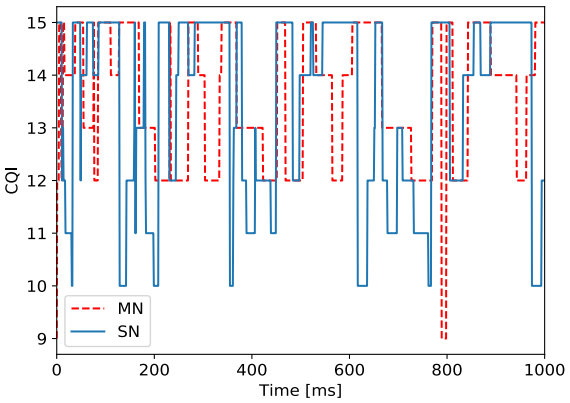


FIGURE 4. CQI values used at the MN and SN in an experiment.

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. Since the *iperf3* runs at the Application layer, it 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 transmitted to the *iperf3 client* via the MN since in our setup, the uplink only uses the mUE-MN connection.

Furthermore, the impact of the PDCP reordering on the obtained aggregate throughput is analyzed by enabling and disabling (NoR) the 3GPP reordering mechanism at the UE. For the former, different values for the PDCP reordering timer ($t_{reordering}$) [10] are evaluated as shown in Table 2. Additionally, the *BH* latency is configured using the Linux traffic control function called NetEm [35]. For this, in the MN's host interface, a fixed delay of 10 ms [16], [36] is added to all the incoming and outcoming packets. Note that the same configuration is also used when the PDCP reordering mechanism is disabled. In this sense, 20 independent evaluations with a duration of 30 seconds each are performed per each analyzed configuration, i.e., per $t_{reordering}$ value, traffic type, and system bandwidths. Table 2 summarizes the different scenario setups evaluated.

TABLE 2. Scenarios evaluated.

Scenario	A	B
System Bandwidth (BW)	$BW_{MN} = 10$ MHz $BW_{SN} = 10$ MHz	$BW_{MN} = 5$ MHz $BW_{SN} = 10$ MHz
PDCP $t_{reordering}$	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

V. RESULTS AND DISCUSSION

Based on the evaluations performed using the framework introduced in Section IV, this section presents and discusses the obtained results in detail. In order 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 illustrates the obtained SC throughput at both BSs and the ideal aggregate throughput for scenarios A and B, both of which serve as a baseline for further comparison.

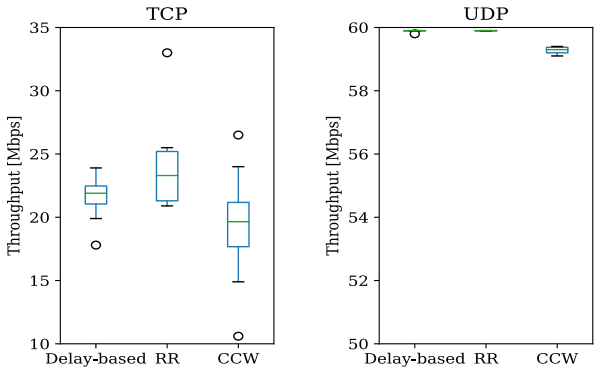
The OAI specifications state that the maximum downlink throughput for UDP traffic in SC operation and highest CQI value is 16-17 Mbps and 34-35 Mbps with 5 MHz and 10 MHz of bandwidth, respectively. [37]. Since we use a real CQI trace, in which the CQI values change on average every 20 ms, the throughput obtained in our testbed experiments and indicated in Table 3 is different to the specified by the OAI. This shows the effect of such CQI variation on the obtained SC throughput.

TABLE 3. Throughput obtained for scenarios A and B with a CQI trace and SC operation.

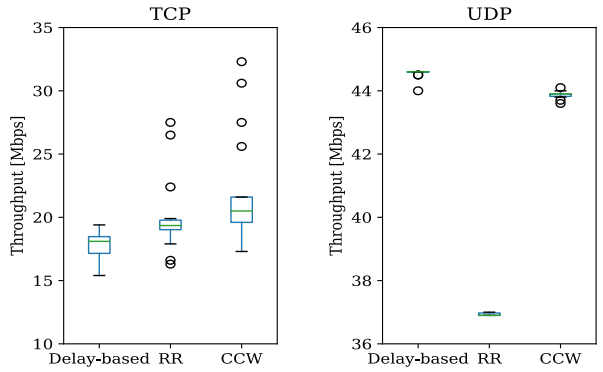
Traffic Type	TCP				UDP			
Scenario	A		B		A		B	
BS Type	MN	SN	MN	SN	MN	SN	MN	SN
Average SC Throughput (Mbps)	28.5	27.5	14.1	27.5	29.9	30	14.6	30
Confidence Interval (95%)	0.26	0.89	0.06	0.89	0.03	0.01	0.02	0.01
Average Ideal Aggregate Throughput (Mbps)	56 ±1.15		41.6 ±0.95		59.9 ±0.04		44.6 ±0.03	

A. 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 illustrated in Figs. 5 and 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 [38]. 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, allow us to easily identify and compare the dispersion of the throughput obtained

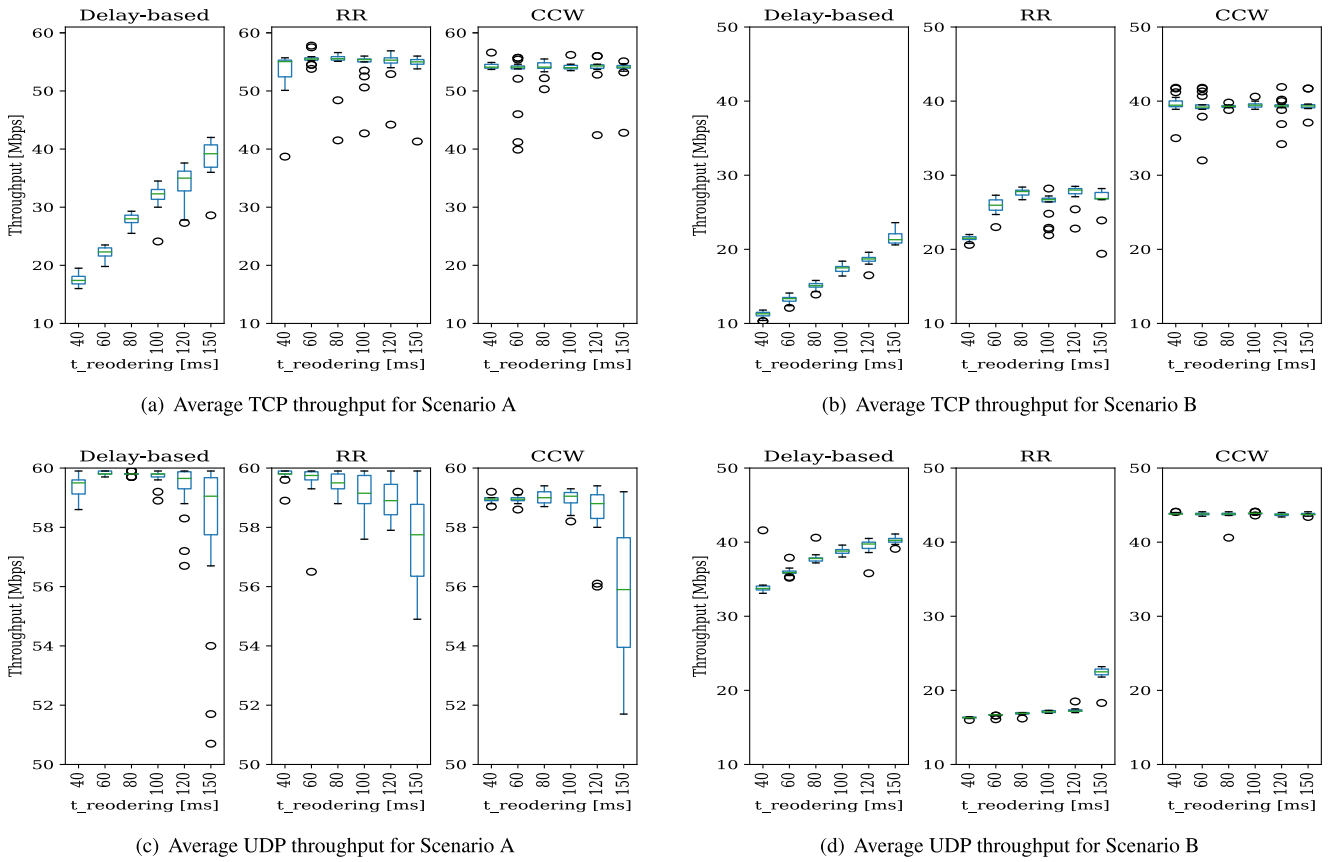


(a) TCP and UDP throughputs for Scenario A and NoR



(b) TCP and UDP throughputs for Scenario B and NoR

FIGURE 5. Aggregate throughput obtained when the reordering mechanism is disabled.



(a) Average TCP throughput for Scenario A

(b) Average TCP throughput for Scenario B

(c) Average UDP throughput for Scenario A

(d) Average UDP throughput for Scenario B

FIGURE 6. **Aggregate throughput for different reordering timeout values, when PDCP reordering is enabled.**

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 reaction of the congestion control mechanisms is determined by the network conditions.

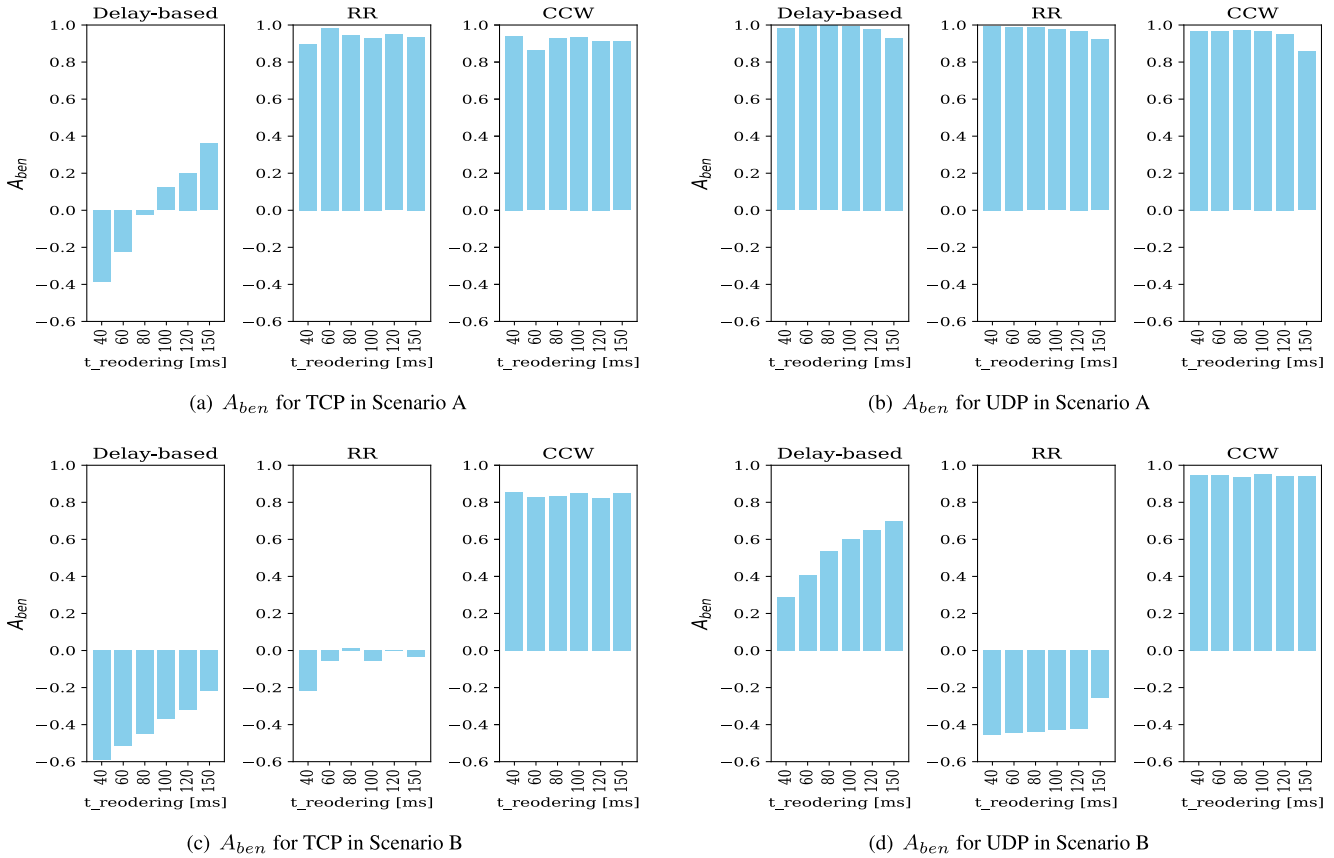
1) REORDERING MECHANISM DISABLED

Figs. 5(a) and 5(b) 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. In fact, they achieve less than the 40% of the ideal aggregate throughput in both evaluated scenarios. Because TCP receives a large amount of out-of-order packets, it continuously reacts with *fast retransmissions*. This reduces the TCP congestion window size and degrades the throughput performance.

For UDP traffic, the three algorithms offer comparable results for Scenario A, where on average they achieve 59 Mbps. Nevertheless, approximately 49%, 48%, and 46% of the total PDUs are received out of order with the RR, Delay-based, and CCW algorithms, respectively. Moreover, for Scenario B, the Delay-based and CCW can deal with the system bandwidth difference and achieve 44 Mbps approximately. However, 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.

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 is able to achieve it regardless of the reordering timeout choice and scenario, as can be observed in Fig. 6. The RR works well only in Scenario A, Fig. 6(a). 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. 6(b). These results reflect the inability of RR to dynamically split the user traffic according to the assigned resources at each BS. This



(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

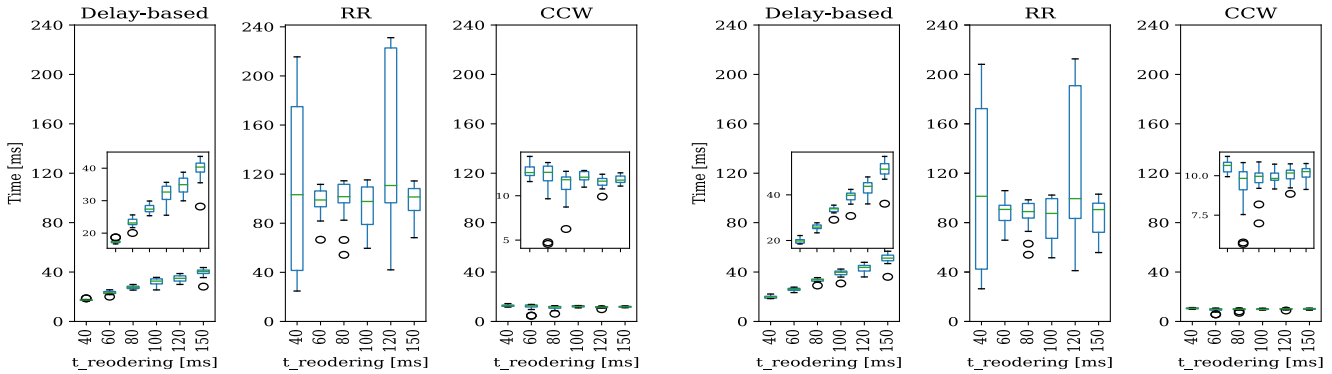
FIGURE 7. Aggregation benefit comparison for Delay-based, RR, and CCW.

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. In this sense, the TCP bursts can make the user traffic be transmitted through a single communication path until the effect of such splitting decisions is known by the MN's PDCP layer. This approach leads to a rapid increase in the delay difference between communication paths, the value of which may be higher than the reordering timeout value configured in the 3GPP reordering mechanism. Hence, packets with discontinuous sequence numbers are delivered to the upper layers after a reordering timeout declaration. Because of this reason, the obtained aggregate throughput with the Delay-based algorithm, depicted in Fig. 6(a) and 6(b), increases with the increment in the reordering timeout value.

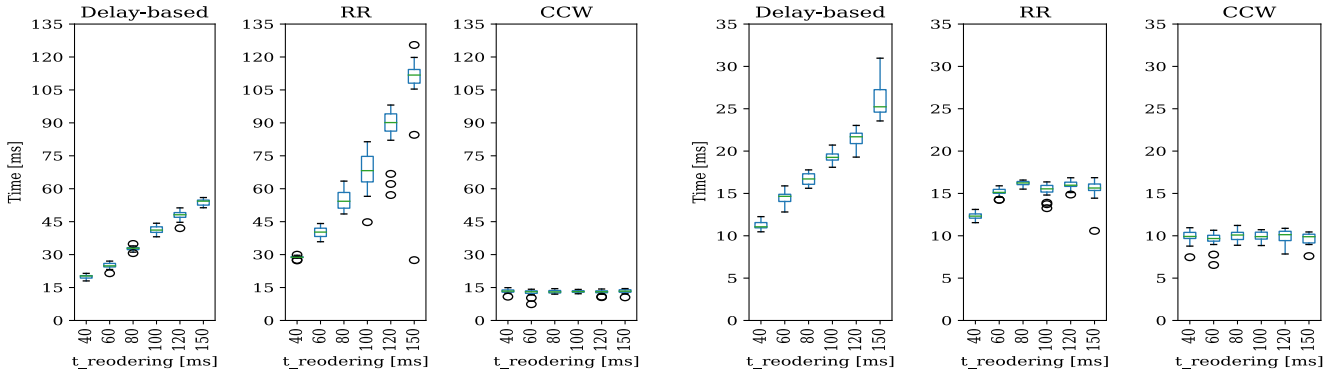
For UDP traffic, Fig. 6(c) shows that the three algorithms obtain, for Scenario A, a similar throughput regardless of the reordering timeout choice. In this case, the continuous traffic pattern of UDP along with the same system bandwidth in both BSs facilitate the traffic splitting for the RR and Delay-based algorithms. For the latter, there are no bursts that can affect the traffic splitting decisions. Additionally, Fig. 6(d) 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 cannot take advantage of the higher SN's bandwidth, and transmit more PDCP PDUs through this communication path. In fact, its simplistic splitting 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 a reordering timeout declaration.

The Delay-based algorithm aims to temporally buffer the PDCP PDUs if the delays of both communication paths are larger than a maximum given value, i.e., d_{max}. When the delay in one of the communication paths is below 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 arrive at the UE after a reordering timeout expiration. Therefore, all the PDUs that arrive at the UE above this time limit are considered lost by the reordering mechanism, and thus, they are not transmitted to the upper layers. Because of this, the aggregated throughput slightly increases with a higher reordering timeout value for the RR and Delay-based algorithms. Note that the d_{max} value taken from [20] and adopted in this study for the Delay-based, may not be optimal in the evaluated scenarios.



(a) MN's RLC delay in Scenario A

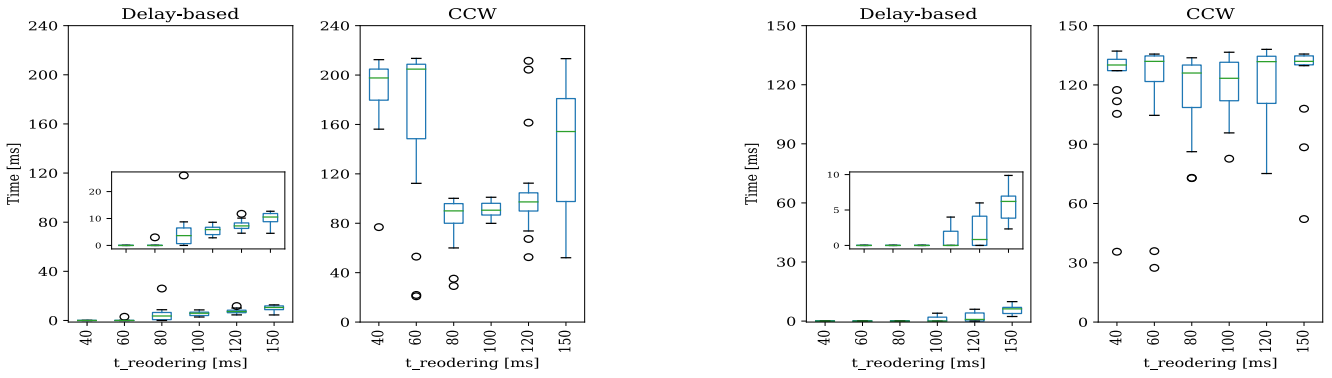
(b) SN's RLC delay in Scenario A



(c) MN's RLC delay in Scenario B

(d) SN's RLC delay in Scenario B

FIGURE 8. **Average RLC sojourn times for TCP traffic using different reordering timeout values.**



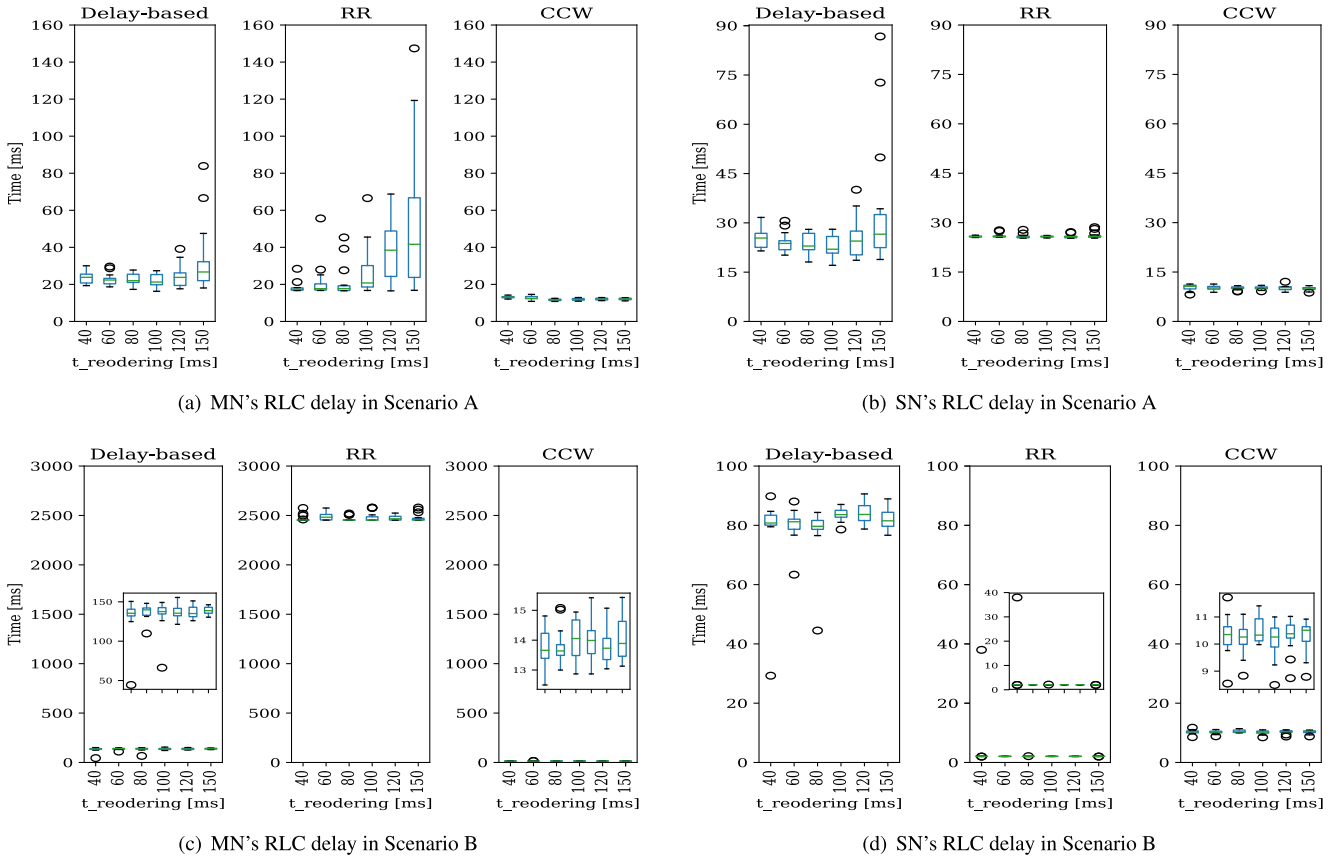
(a) PDCP delay in Scenario A

(b) PDCP delay in Scenario B

FIGURE 9. **Average PDCP sojourn times for TCP traffic using different reordering timeout values.**

Furthermore, as shown in Fig. 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 system bandwidth combination. This is an important characteristic that makes our algorithm robust enough to adapt to the different conditions expected in a mobile network. For a better illustration of the CCW's robustness, we analyze the aggregation benefit A_{ben} obtained with each flow control algorithm.

The aggregation benefit function A_{ben} helps us to quantify and compare the efficiency of each flow control algorithm to aggregate traffic in MR-DC. In this sense, Fig. 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



(a) MN's RLC delay in Scenario A

(b) SN's RLC delay in Scenario A

(c) MN's RLC delay in Scenario B

(d) SN's RLC delay in Scenario B

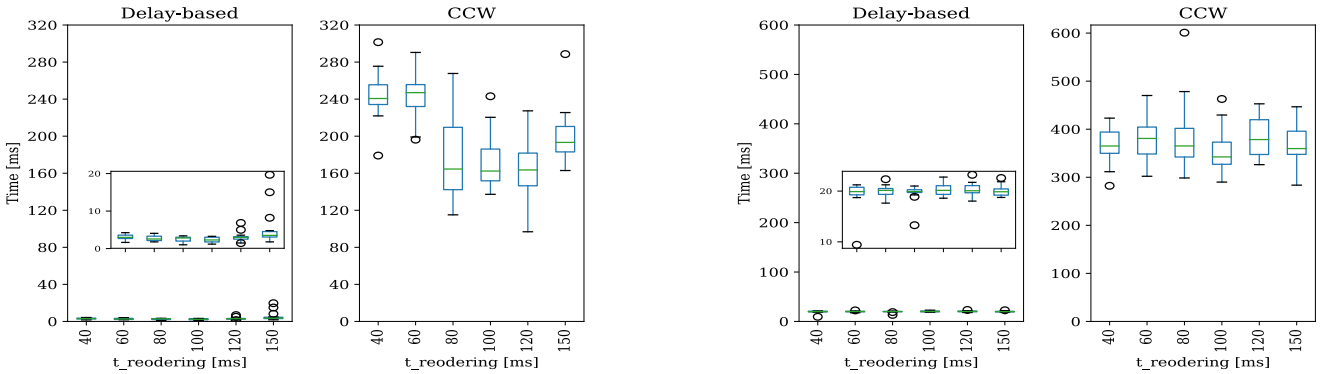
FIGURE 10. Average RLC sojourn times for UDP traffic using different reordering timeout values.

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. 7(a) and 7(b). However, as illustrated in Figs. 7(c) and 7(d) 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. On the contrary, the obtained aggregate throughput for TCP traffic is even lower than the throughput obtained in SC operation.

B. 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 the performance of TCP, as we previously mentioned and observed in Figs. 5(a) and 5(b). 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. Additionally, the sojourn time experienced in the PDCP flow control buffer is also shown for the Delay-based and CCW algorithms. For TCP traffic, Fig. 8 shows that the CCW effectively limits the buffering delay of the RLC buffers in both scenarios. Therefore, it is possible to maintain a continuous data flow towards the UE without significantly increasing the time difference between the communication paths. Even though the use of the CCW's flow control buffer, B_{FC}, increases the packet sojourn time, as depicted in Figs. 9(a) and 9(b), 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.

For UDP traffic, the CCW also maintains lower levels of the RLC buffering delay in both RLC buffers, as shown in Fig. 10, in comparison with the delay obtained with the other two algorithms. Consequently, the CCW can achieve at least 95% of the ideal aggregate throughput regardless of the reordering timeout choice and scenario. Nevertheless, Fig. 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



(a) PDCP delay in Scenario A

(b) PDCP delay in Scenario B

FIGURE 11. Average PDCP sojourn times for UDP traffic using different reordering timeout values.

the throughput in both BSs is significantly different, like in Scenario B, the CCW notoriously obtains a higher aggregate throughput than the one obtained with the Delay-based and RR algorithms.

Furthermore, the results illustrated in Figs. 10(c) and 10(d) for Scenario B indicate that the CCW keeps on average 4 ms of buffering delay 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 than the other. Since the 3GPP reordering mechanism is not able to deal with such delay differences, the performance of the algorithms mentioned above is significantly lower than the CCW. Because of this reason the aggregate throughput obtained with the Delay-based and RR algorithms slightly increases using higher reordering timeout values. Note that the PDUs may have to wait longer in the reordering buffer with higher reordering timeout values, which may cause bufferbloat. It is important to remark that our CCW flow control algorithm aims not 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.

VI. CONCLUSION

In this article we have 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 in use. 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.

Our testbed experiments showed that the TCP traffic performance might be seriously degraded by the out-of-order arrival of PDCP PDUs in MR-DC operation. However, if properly configured, the PDCP reordering mechanism that should be employed at the PDCP receiver for MR-DC operation alleviates this issue. 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 sense, 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 available 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.

Furthermore, 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 also achieve such performance, but only in Scenario A. For Scenario B, both algorithms offer a more inferior aggregation benefit.

Because of the considerations mentioned above, we have demonstrated that the CCW is an effective flow control mechanism to aggregate TCP- and UDP-based traffic, which can be implemented in any MR-DC architecture option.

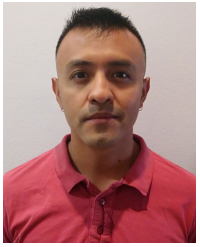
Finally, as part of our future work, we plan to extend the design of the CCW algorithm to operate in BSs that use different TTI/slot durations. Additionally, it is crucial to reduce the delay added by the flow control mechanism so that MR-DC can also be used for applications that simultaneously demand low-latency and high-throughput.

REFERENCES

[1] *5G; Study on Scenarios and Requirements for Next Generation Access Technologies (Release 16)*, 3GPP, document TR 38.913, 2020, pp. 1–42.

[2] C. Rosa, K. Pedersen, H. Wang, P. H. Michaelsen, S. Barbera, E. Malkamäki, T. Henttonen, and B. Sébire, "Dual connectivity for LTE small cell evolution: Functionality and performance aspects," *IEEE Commun. Mag.*, vol. 54, no. 6, pp. 137–143, Jun. 2016.

[3] *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall Description; Stage 2*, 3GPP, document TS 36.300–V16.5.0, 2021.
[4] *Universal Mobile Telecommunications System (UMTS); LTE; 5G; NR; Multi-connectivity; Overall description; Stage-2*, 3GPP, document TS 37.340 - V16.5.0, 2021.
[5] *Study on Small Cell Enhancements for EUTRA and EUTRAN Higher Layer Aspects*, 3GPP, document TR 36.842, 2013.
[6] 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.
[7] A. A. Laghari, H. He, and M. I. Channa, "Measuring effect of packet reordering on quality of experience (QoE) in video streaming," *3D Res.*, vol. 9, no. 3, Sep. 2018.
[8] D. Kaspar, "Multipath aggregation of heterogeneous access networks," Ph.D. dissertation, Dept. Math. Natural Sci., Univ. Oslo, Oslo, Norway, 2012.
[9] D. Raca, J. J. Quinlan, A. H. Zahran, and C. J. Sreenan, "Beyond throughput: A 4G LTE dataset with channel and context metrics," in *Proc. 9th ACM Multimedia Syst. Conf. (MMSys)*, New York, NY, USA, Jun. 2018, pp. 460–465.
[10] *5G; NR; Packet Data Convergence Protocol (PDCP) Specification (Release 15)*, 3GPP, document TS 38.323–V16.4.0, 2021, p. 27.
[11] M. Irazabal, E. Lopez-Aguilera, I. Demirkol, R. Schmidt, and N. Nikaein, "Preventing RLC buffer sojourn delays in 5G," *IEEE Access*, vol. 9, pp. 39466–39488, 2021.
[12] Intel, *Contribution: Throughput Evaluation and Comparison of With and Without UP Bearer Split*, Intel Corporation, 3GPP, document R2-132859, Aug. 2013.
[13] H. Wang, C. Rosa, and K. I. Pedersen, "Dual connectivity for LTE-Advanced heterogeneous networks," *Wireless Netw.*, vol. 22, no. 4, pp. 1315–1328, May 2016.
[14] M.-S. Pan, T. M. Lin, C. Y. Chiu, and C. Y. Wang, "Downlink traffic scheduling for LTE-A small cell networks with dual connectivity enhancement," *IEEE Commun. Lett.*, vol. 20, no. 4, pp. 796–799, Apr. 2016.
[15] K. Nguyen, M. G. Kibria, J. Hui, K. Ishizu, and F. Kojima, "Minimum latency and optimal traffic partition in 5G small cell networks," in *Proc. IEEE 87th Veh. Technol. Conf. (VTC Spring)*, Jun. 2018, pp. 1–5.
[16] R. P. Antonioli, E. B. Rodrigues, D. A. Sousa, I. M. Guerreiro, C. F. M. e Silva, and F. R. P. Cavalcanti, "Adaptive bearer split control for 5G multi-RAT scenarios with dual connectivity," *Comput. Netw.*, vol. 161, pp. 183–196, Oct. 2019.
[17] R. P. Antonioli, I. M. Guerreiro, D. A. Sousa, E. B. Rodrigues, C. F. M. e Silva, T. F. Maciel, and F. R. P. Cavalcanti, "User-assisted bearer split control for dual connectivity in multi-RAT 5G networks," *Wireless Netw.*, vol. 26, no. 5, pp. 3675–3685, Jul. 2020.
[18] Y. Wang, C. Sun, F. Jiang, and J. Jiang, "Blocking- and delay-aware flow control using Markov decision process," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Aug. 2020, pp. 905–910.
[19] B. Jin, S. Kim, D. Yun, H. Lee, W. Kim, and Y. Yi, "Aggregating LTE and Wi-Fi: ToWARD iNTRA-cELL fAIRNESS AND hIgh TCP performance," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6295–6308, Oct. 2017.
[20] D. López-Pérez, D. Laselva, E. Wallmeier, P. Purovesi, P. Lunden, E. Virtej, P. Lechowicz, E. Malkamaki, and M. Ding, "Long term evolution-wireless local area network aggregation flow control," *IEEE Access*, vol. 4, pp. 9860–9869, 2016.
[21] *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Layer Procedures*, 3GPP, document TS 36.213-V16.6.0, 2021.
[22] *5G; NR; Physical Layer Procedures for Data*, 3GPP, document TS 38.214–V16.6.0, 2021.
[23] M. M. Nasralla, N. Khan, and M. G. Martini, "Content-aware downlink scheduling for LTE wireless systems: A survey and performance comparison of key approaches," *Comput. Commun.*, vol. 130, pp. 78–100, Oct. 2018.
[24] S. Ahmadi, *LTE-Advance: A Practical Systems Approach 3GPP LTE Releases 10 and 11 Radio Access Technologies LTE-Advanced*, 1st ed. Amsterdam, The Netherlands: Elsevier, 2014.
[25] *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) Protocol Specification*, 3GPP, document TS 36.321-V16.5.0, 2021, p. 37.
[26] *NR; Medium Access Control (MAC) Protocol Specification*, 3GPP, document TS 38.321-V16.5.0, 2021, p. 79.
[27] A. Misra, T. Ott, and J. Baras, "Effect of exponential averaging on the variability of a RED queue," in *Proc. IEEE Int. Conf. Commun. Conf. Rec.*, vol. 6, Jun. 2001, pp. 1817–1823.
[28] L. Kleinrock, *Queuing Systems: Theory*. vol. 1. Hoboken, NJ, USA: Wiley, 1975.
[29] *5G; NG-RAN; NR User Plane Protocol*, 3GPP, document TS 38.425-V16.3.0, 2021.
[30] L. Kleinrock, "Internet congestion control using the power metric: Keep the pipe just full, but no fuller," *Ad Hoc Netw.*, vol. 80, pp. 142–157, Nov. 2018.
[31] *NR; Radio Link Control (RLC) Protocol Specification*, 3GPP, document TS 38.322-V16.2.0, 2021, p. 34.
[32] N. Nikaein, M. K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "OpenAirInterface: A flexible platform for 5G research," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 5, pp. 33–38, 2014.
[33] *Open-Access Research Testbed for Next-Generation Wireless Networks (ORBIT)*. Accessed: May 5, 2021. [Online]. Available: https://www.orbit-lab.org/
[34] K. Salah and F. Haidari, "On the performance of a simple packet rate estimator," in *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl.*, Mar. 2008, pp. 392–395.
[35] S. Hemminger, "Network emulation with NetEm," in *Proc. Aust. 6th Nat. Linux Conf.*, Canberra, ACT, Australia, Apr. 2005.
[36] *LTE; Scenarios and Requirements for Small Cell Enhancements for E-UTRA and E-UTRAN*, 3GPP, document TR 36.932-V12.1.0, 2014, p. 15.
[37] F. Kaltenberger, "OpenAirInterface 5G overview, installation, usage," in *Proc. OpenAirInterface Workshop*, Beijing, China, 2019.
[38] D. F. Williamson, R. A. Parker, and J. S. Kendrick, "The box plot: A simple visual method to interpret data," *Ann. Internal Med.*, vol. 110, no. 11, pp. 916–921, 1989.



CARLOS PUPIALES received the B.Sc. degree from Escuela Politécnica Nacional, Ecuador, and the M.Sc. degree in telecommunications engineering from the University of Melbourne, Australia. He is currently pursuing the Ph.D. degree with the Department of Network Engineering, Universitat Politècnica de Catalunya, Spain. His main research interest includes the area of network protocols in mobile networks. He was a recipient of the Best Demo Award in IEEE MASS 2019.

DANIELA LASELVA received the M.Sc. degree in electrical engineering from the Polytechnic of Bari, Italy, in 2002. From 2002 to 2006, she was active in COST273 and EU FP6 Project WINNER on MIMO radio channel modeling with Elektrobit, Finland, and worked as a Senior Design Engineer with Nokia, Finland. Since 2006, she has been with Nokia Aalborg, Denmark. She is currently a Senior Research Specialist with the Nokia Standardization and Research Team. At present, she is engaged with 5G new radio design and standardization, including solutions for efficient transmission of small data, UE power saving, multi-connectivity, and support of mission-critical applications. She has coauthored more than 30 peer-reviewed publications and four book chapters, and she is an inventor of numerous patents on a wide range of topics.

ILKER DEMIRKOL (Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in computer engineering from Boğaziçi University, İstanbul, Turkey. He is currently an Associate Professor with the Department of Mining, Industrial and ICT Engineering, Universitat Politècnica de Catalunya, where he works on wireless networks and wake-up radio systems. His research interests include communication protocol development for the aforementioned networks, along with performance evaluation and optimization of such systems. He was a recipient of the 2010 Best Mentor Award from the Electrical and Computer Engineering Department, University of Rochester, Rochester, NY, USA.