Impact of Packetization and Scheduling on C-RAN Fronthaul Performance

Chia-Yu Chang, Navid Nikaein, Thrasyvoulos Spyropoulos
Communication Systems Department
EURECOM, Biot, France 06410
Email: firstname.lastname@eurecom.fr

***Abstract*—Being considered as a key enabler for beyond 4G networks, Cloud-RAN (CRAN) offers advanced cooperation and coordinated processing capabilities and brings multiplexing gains. The high capacity and low latency fronthaul (FH) links requirement in the CRAN architecture can be reduced by a flexible functional split of baseband processing between remote radio units (RRUs) and Baseband units (BBUs). Under the wide adoption of Ethernet in data centers and the core network, we consider the Radio over Ethernet (RoE) as an off-the-shelf alternative for FH link in this work. Moreover, the packetization process that packs each sample into Ethernet packets transported over the FH link will impact the CRAN performance. To this end, we investigate the impact of packetization on the proposed CRAN network and provide a packetization algorithm over the FH links. Furthermore, we also survey and analyze various packet scheduling policies applied at the aggregated RRU gateway in order to increase the multiplexing gain. Finally, the simulation results provide more in-depth insights on the potential multiplexing gains in terms of the maximum number of RRUs that can be supported over the Ethernet-based FH network.**

I. Introduction

Toward the rapid increasing of the user traffic in the future radio access network (RAN), the deployment of new base stations (BSs) is critical for user satisfaction. A huge investment in capital expenditure and operating expense is expected under conventional RAN. Applying the Cloud/Centralized RAN (C-RAN) [1] architecture can largely reduce the expenditure and it is considered as one of the most promising technologies that will reshape the future 5G architecture. Unlike conventional RAN, C-RAN decouples the Baseband units (BBU) from the remote radio head (RRH), also known as remote radio unit (RRU), at the network edge. The baseband processing is now centralized into a single pool of shared and dynamically allocated BBUs, offering energy efficiency and multiplexing gains. These BBU functions can be implemented on commodity hardware and performed on virtual machines, further benefiting from softwarization and Network Function Virtualization (NFV). Finally, the centralized BBU functions facilitate coordinated multi-point processing.

Despite its appeal, one key obstacle in the adoption of the C-RAN architecture is the excessive capacity and latency requirements on the Fronthaul (FH) link connecting an RRU with the BBU cloud. An example in [2] shows shifting all baseband processing to a remote cloud implies that to support a 75 Mbps user data rate, for a single user, approximately 1 Gbps of information is needed to transport on the FH link. Furthermore, the user expects to receive an ACK/NACK response within 4ms [3] after its transmission, imposing a strong latency requirement on the FH link.

In order to relax the excessive FH capacity constraint, the concept of C-RAN is revisited, and a more flexible distribution of baseband functionality between the RRUs and the BBU pool is considered [4]. Rather than offloading all the baseband processing on the BBU pool, it is possible to keep a subset of these blocks in the RRU. This concept is also known as *Flexible Centralization*. By gradually placing more and more BBU processing (e.g., Cyclic prefix and guard band removal, Unused resource element removal, etc.) at RRUs, the FH capacity requirement becomes smaller. Nevertheless, flexible centralization needs more complex and more expensive RRU and it reduces the opportunities for coordinated signal processing and advanced interference avoidance schemes. Consequently, flexible or partial centralization is a trade-off between what is gained in terms of FH requirements and what is lost in terms of C-RAN features.

Another key question is how the information between the RRU and BBU is transported over the FH link. A number of FH transmission protocols are under investigation, such as CPRI [5], OBSAI [6] and ORI [7]. However, these have mainly been considered for carrying raw I/Q samples in a fully centralized C-RAN architecture. In light of the different possible functional splits, different types of information are transported over the FH link. Given the extensive adoption of Ethernet in clouds, data centers, and the core network, Radio over Ethernet (RoE) [8] could be a generic, cost-effective, off-the-shelf alternative for FH transport. Further, while a single FH link per RRU, all the way to the BBU pool, has usually been assumed, it is expected that the FH network will evolve to a more complex multihop mesh network topology, requiring switching and aggregation [9], such as the one in Fig. 1. This is facilitated by applying a standard Ethernet approach and SDN-based switching capabilities.

Nevertheless, packetization over the FH introduces additional concerns related to latency and overhead. As information arriving at the RRU and/or BBU needs to be inserted in an Ethernet frame, header-related overhead is introduced per frame. To ensure this overhead is small, and does not waste the potential capacity gains from functional splitting, it would thus be desirable to fill all the payload before sending. However, waiting to fill a payload, introduces additional latency, possibly using up some of the latency budget as explained earlier. Hence, it is important to consider the impact of packetization on the FH capacity and latency performance to understand the feasibility and potential gains of different approaches.

Moreover, all RRUs are aggregated and multiplexed at the RRU gateway where the packets are switched and/or routed to

RRU/BBU in Fig. 1. The packet scheduling policy at the RRU gateway impacts the latency and fairness for all connected RRUs. In view of the fairness, all RRUs expect to have the same level of latency irrelevant to its traffic characteristic. However, the unification of the latency implies less variability and flexibility on the packet scheduling and degrades the multiplexing benefit. Therefore, the packet scheduling is surveyed to have a understanding of the trade-off between fairness for different RRUs and the multiplexing gains. In summary, the main contributions of this work are the following:

1) We survey several packetization techniques and provide the packetization algorithm for C-RAN network;
2) We then analyze the impact of packet scheduling policies at the RRU gateway where the packets are switched;
3) Finally, we use our results to identify different packetization and scheduling policies, and provide insights on the achievable gain in terms of the supported RRU number of FH capacity-limited problem;

The rest of this paper is organized as follows. Sec. II presents some related works. In sec. III, we introduce the considered C-RAN network topology, possible functional splits and the considered problem based on LTE HARQ timing constraint. Sec. IV focuses on the impact of packetization. Sec. V surveys the packet scheduling methods. Sec. VI provides simulation results to validate the impact of packetization and packet scheduling. Finally, sec. VII concludes the paper.

II. Related work

Recently, several standardization activities are redefining the FH network towards a packet-based architecture. The goal is to design a variable rate, multipoint-to-multipoint, packet-based FH interface. Ref. [10] presents the Ethernet-transported Next Generation Fronthaul Interface (NGFI) and its design principles, application scenarios, and network measurement results. IEEE 1904.3 specifies the RoE encapsulations and mappings in [8]. Fronthaul and backhaul requirements, transmission technologies are discussed in [4] and the FH rate per each split under a specific configuration is provided. Ref. [11] provides the FH data rate CDF and the number of supported RRUs per transmission technology with and without considering the multiplexing gain (calculated from central limit theorem). Per-split strategies are elaborated in [12] to reduce the FH requirements while maintaining centralization advantages. Bandwidth reduction is analyzed in [13] under the split-PHY processing architecture. Our previous work in [2] provided the peak rate analysis on different splits and the joint packetization-split survey; however, in this work, we further consider the time-multiplexing impact and provide the packetization algorithm and the packet scheduling policy. To sum up, this work complements above existing studies in that it analyzes the impact of packetization and packet scheduling policy in the future packet-based FH network.

III. System Model

A. C-RAN Network Topology

The initial C-RAN concept is to apply a single direct FH link to connect each RRU to the BBU pool. However, due to concerns to scalability, capital expenditure, and multiplexing, it is expected that the FH will evolve towards more complex, shared topologies, similar to the backhaul network [14]. In this work, we focus our discussion on a simple topology, which is characteristic of this envisioned evolution in Fig. 1 with N RRUs. The considered topology provides a simple but a realistic deployment of the mesh C-RAN network.

After some baseband processing (based on functional split) at the i^{th} RRU ($i \in [1, N]$), samples are packetized into Ethernet packets. These packets are then queued at the RRU for transmission until FH segment I is available. The distance and capacity of the FH segment I between the i^{th} RRU and the RRU gateway are denoted as d_i and r_i. All connected RRUs are aggregated and multiplexed/demultiplexed at the RRU gateway where the packets are switched and routed between RRUs and the BBU pool. Moreover, the RRU gateway can schedule prioritized packet transmission on FH segment II and discard packets that violate HARQ timing constraint. The single-machine, multi-queue model is applied at the RRU gateway, and the packets from different RRUs are stored in each first-in-first-out (FIFO) queue. Then, based on the packet scheduling algorithm, packets are transported over FH segment II. The distance and capacity of the FH segment II are denoted as D and R. Afterwards, the rest of the baseband processing is done by the BBU after the arrival of packets.

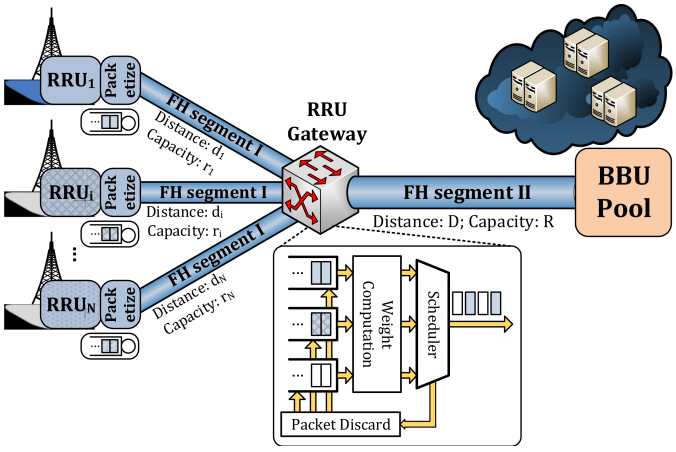

Fig. 1: Considered C-RAN network topology

Without loss of generality, we apply the capacity assumption as in [2] as $r_i = r = 4\,Gbps$, $\forall i \in [1, N]$ on FH segment I and $R = 20\,Gbps$ on FH segment II. Moreover, the location of the RRU gateway is assumed at the central of the network to save the total FH segment I expenditure. The distance between the BBU pool and RRU gateway is assumed to be $D = 10\,km$ which is the maximum value of most deployment [15].

B. Split over uplink functions

The functional split between RRU and BBU affects the experienced FH rate and latency. In Fig. 2, we apply the same five uplink physical layer (PHY) functional splits (Split A, B, C, D, E) in [2]. Nevertheless, some splits with the constant data rate (Split A and B) does not provide any multiplexing gain over the FH links. The multiplexing gain can only be exploited when the UE-domain processing is involved at the RRU for Split C, D and E. While different RRU-BBU PHY functional splits may be applied to different RRUs; however, we assume all supported RRUs apply the same PHY function split for simplicity.

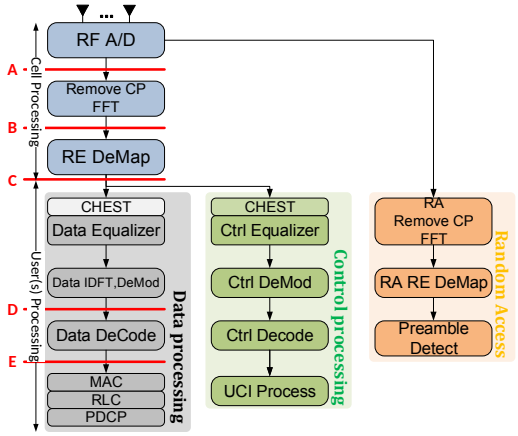

Fig. 2: PHY functional splits on LTE uplink [2]

C. General HARQ timing constraint

The HARQ timing constraint is to require every received MAC protocol data units (PDUs) should be acknowledged (ACK'ed) or non-acknowledged (NACK'ed) within a specific time duration. In FDD LTE case, the HARQ round trip time (T_{HARQ}) is 8ms and the uplink HARQ timing is in Fig. 3. Each MAC PDU sent at $(N)^{th}$ subframe is propagated (T_{prop}), acquired (T_{acq}), and processed both in reception ($T_{Rx,eNB}$) and transmission ($T_{Tx,eNB}$) chains for the ACK/NACK response. Then, this response is received at $(N+4)^{th}$ subframe by UE and the re-transmission starts at $(N+8)^{th}$ subframe.

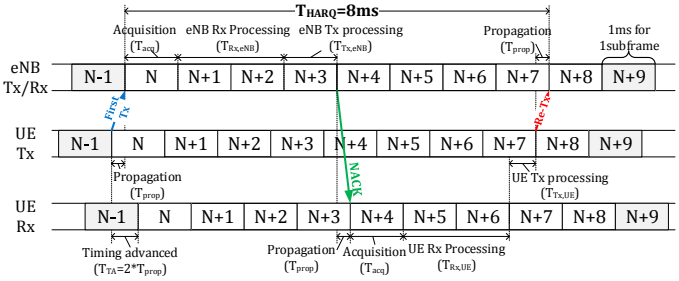

Fig. 3: Uplink HARQ timing

Based on Fig. 3, the maximum allowed time for eNB reception and transmission is in Eq. (1). The acquisition time (T_{Acq}) to acquire all samples in a subframe takes 1ms. Though the transmission chain needs to wait the reception chain for the ACK/NACK response; however, most of its processing can be prepared before the end of reception chain. In this sense, the processing time of the transmission chain ($T_{Tx,eNB}$) is independent from the reception chain processing time and can be bounded irrelevant to the ACK/NACK response. In hence, we assume the maximum processing time of transmission chain is 1ms as [3], and the constraint is reformulated in Eq. (2). Note this calculation is based on transmission time interval (TTI) equals to 1ms.

$$T_{Rx,eNB} + T_{Tx,eNB} \leq \frac{T_{HARQ}}{2} - T_{Acq} = 3ms \tag{1}$$

$$T_{Rx,eNB} \leq 3ms - max\left(T_{Tx,eNB}\right) = 2ms \tag{2}$$

D. Considered problem formulation

The maximum reception chain processing time in Eq. (2) can be further decomposed into overall baseband processing time (T_{proc}) and FH delay (T_{FH}) in C-RAN. The problem to support as more RRUs as possible in the FH capacity-limited C-RAN scenario is then formulated in Eq. (3). All components of the FH delay (T_{FH}) are further explained in TABLE I. For the i^{th} RRU of all N supported RRUs, the last packet ($k = n_i(j)$) at each j^{th} time-critical symbol in a TTI shall fulfill the constraint in Eq. (3) where $n_i(j)$ is the total number of packets at the j^{th} symbol of i^{th} RRU. The time-critical symbols are the ones used to initialize the baseband processing at BBU, e.g., all symbols for split A and B, the last reference signal (RS) symbols and non-RS symbol of each slot used for the demodulation at BBU for split C, the last demodulated/decoded symbol of each TTI for the decode/MAC process at BBU for split D/E. The overall baseband processing time T_{proc} is the sum of the baseband processing time at RRU and BBU, and it is a function of the PRB number, modulation and coding scheme (MCS) index and virtualization platform [3]. For simplicity, we assume all RRUs and BBU have the same virtualization platform that takes the longest processing time (DOCKER in [3]) and apply the maximum PRB number (100PRB), the highest MCS index (27) to get $max\left(T_{proc}\right)$ as the upper bound of T_{proc}.

$$\begin{aligned} &max\ N \\ &s.t.\ T_{FH}(i) \\ &\quad = T_{pkt}(i,k) + T_{Q_R}(i,k) + T_{p1}(i) + T_{Q_G}(i,k) + T_{p2} \\ &\quad \leq 2ms - max\left(T_{proc}\left(PRB, MCS, Platform\right)\right), \\ &\forall i \in [1,N], j \in \{\text{Time-critical symbols}\}, k = n_i(j) \end{aligned} \tag{3}$$

IV. Impact of Packetization

Packetization aims to form Ethernet packets from the incoming samples after baseband processing. In the following subsections, we firstly introduce the joint impact of packetization and baseband processing on different splits, then we survey the impact of maximum payload size and finally provide the packetization algorithm.

A. Joint impact of packetization and baseband processing

For the time-critical symbols, the decision to packetize samples immediately or with more samples from next symbol shall be made. On one hand, it takes $T_s \approx 71.354\mu s$ to wait for samples from next symbol; on the other hand, 78 bytes of overhead are generated for one extra packet. Considering the trade-off, we provide the zero cross-symbol bits condition in Eq. (4) to make the decision for time-critical symbols packetization. If the condition is fulfilled, then the samples are packetized immediately. The $E[x_{i,j}]$ is the expected number of samples in bytes from the j^{th} symbol of i^{th} RRU, X is the packet payload size in bytes, and $\lceil . \rceil$ is ceiling function.

$$\begin{aligned} &T_s \cdot min\left(r, \frac{R}{N}\right) - 8 \cdot \left(E[x_{i,j}] + 78\left\lceil \frac{E[x_{i,j}]}{X} \right\rceil\right) > 0 \\ &\forall i \in [1,N], j \in \{\text{Time-critical symbols}\} \end{aligned} \tag{4}$$

Based on the FH link capacity values (r, R) in Sec. III-A and the RRU number that is less than the maximum supported RRU number in Sec. VI; this condition is fulfilled for all splits. Thus, samples of time-critical symbols are packetized immediately ($T_{pkt} = 0$). In contrast, samples of symbols that are not time-critical do not need to be packetized immediately since there is no delay constraints on them. In following parts, we further introduce the packetization impact on each split.

TABLE I: Fronthaul delay components

Component	Notation	Description
Packetization delay	T_{pkt}	The time required to packetize samples into Ethernet packets
RRU queuing delay	T_{Q_R}	The time interval from the packet is packetized until it is transported over FH segment I
FH segment I propagation delay	T_{p_1}	The time required to transport packet over the FH segment I, it equals to d_i/c where c is the light speed
RRU gateway queuing delay	T_{Q_G}	The time interval from the packet arrived at the RRU gateway until it is transported over FH segment II
FH segment II propagation delay	T_{p_2}	The time required to transport packet over the FH segment II, it equals to D/c where c is the light speed

1) Split A, B, C: Packets are formed immediately if incoming bits belong to time-critical symbols defined in Sec. III-D. Otherwise, packets are formed until payload is full.

2) Split D: The sample output time is highly impacted by the channel estimation and demodulation operation in Fig. 4. The first step is to do the channel estimation on the pre-allocated RS symbols, and these results are used to interpolate the channel estimation for non-RS symbols. A common factor here is the **look-ahead depth** which refers the non-causality depth of the future RS symbols that are used to interpolate the channel estimation of the non-RS symbol. For instance, the 7^{th} symbol channel estimation in Fig. 4 is interpolated from the 4^{th} and 11^{th} RS symbol so the look-ahead depth is $11-7=4$. After the channel estimation, the non-RS symbols are equalized, demodulated and packetized based on its operation period. Possible data channel look-ahead depths of both cyclic prefix (CP) types in two different operation periods are in TABLE II.

TABLE II: Possible look-ahead depth

CP type \ Operation periods	Subframe	Slot
Normal CP	4, 5, 6, 8, 9, 10	3
Extended CP	3, 4, 5, 7, 8	2

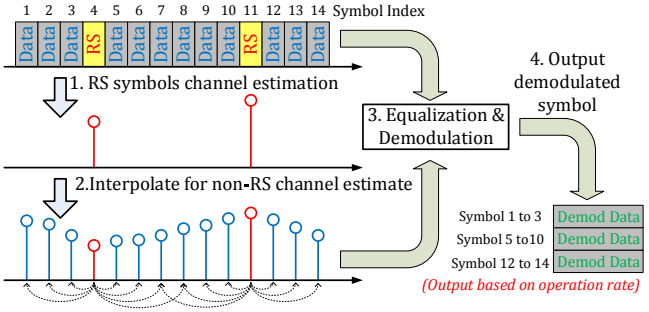

Fig. 4: Channel estimation and demodulation operation

One significant characteristic of this split is its bursty traffic at the end of the operation period (subframe or slot), and it imposes a heavy burden on the FH links since packets will be flooded and the traffic is congested in this duration. In hence, the **pre-fetch** scheme is proposed which tends to demodulate and packetize samples once all required RS symbols within the look-ahead depth are received (Step 3a in Fig. 5) rather than following normal operation period (Step 3b in Fig. 5). While the pre-fetch scheme increases the overhead, it effectively mitigates the bursty traffic congestion. Further, the pre-fetch scheme can also be applied to the control channel; however, little impact is observed because control channel has fewer number of samples and is less congested.

3) Split E: The bit-rate processing is done at the RRU for both control and data channel, and the traffic is always bursty at the end of TTI. To reduce the delay of instantaneous control information (ACK/NACK, Channel state information, etc.), these samples can be packetized immediately.

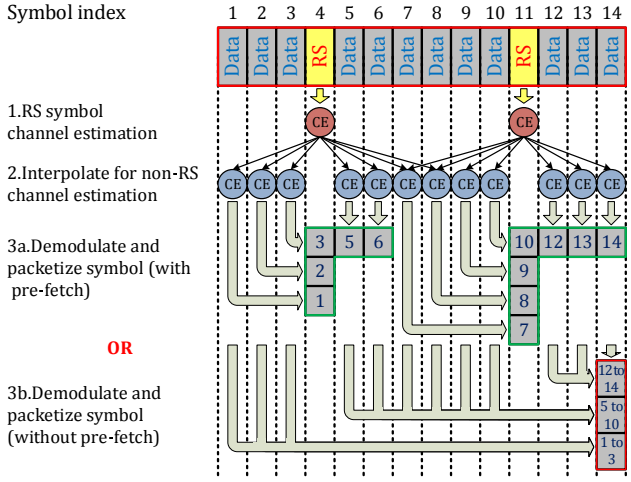

Fig. 5: Pre-fetch on the demodulated symbols

B. Impact of maximum payload size

Using a smaller payload size can replace a large packet with few small packets, so the RRU queuing delay is decreased but with extra overhead. Considering this trade-off, the optimal payload size that minimizes the FH delay (T_{FH}) in Eq. (3) for each split can be known after exhaustive simulations; however, a simple way to approximate it is proposed when the FH link is near fully-loaded. Denote the maximum payload size as X bytes which is less than the maximum payload size of a jumbo frame (J), and we aim to find the value of X that makes the derivative of maximum FH delay 0 in Eq. (5). Only T_{Q_R} and T_{Q_G} are related to the value of X, and we formulate the sum of T_{Q_R} and T_{Q_G} as the sum of the first packet transportation time on FH segment I and the serialization time of all packets on FH segment II as FH link is near fully-loaded. The $E[x_{i,j}]$ is defined in IV-A, and $E[x_i]=\sum E[x_{i,j}]$ is the sum of the expected number of samples for all symbols in the almost fully-loaded FH duration and $M_i=\lceil E[x_i]/X \rceil$ is the expected number of packets in this duration.

Considering the condition that the number of output samples of the first symbol is much larger than the maximum payload size of a jumbo frame (e.g., $E[x_{i,1}] \gg J$), and we can get: $X=min(X, E[x_{i,1}])$ and $M_i \approx \tilde{M_i}=(E[x_i]/X)$. Then, we define $T=\sum_{i=1}^{N} E[x_i]$ and get the optimal payload size that makes the derivative equals to 0. Before showing the result, we review two exceptional conditions. For the split that does not fulfill the condition (Split E, split D with small look-ahead or pre-fetch), $T_{Q_{RRU}}$ is not related to the payload size X, so the derivative in Eq. (5) is negative and the optimal payload size is the maximum jumbo frame payload size (J). Moreover, for splits with higher rate (Split A, B), the payload size shall be larger than a lower bound (X_{lb}) to avoid overflow condition in the FH link caused by excessive overhead in each subframe duration (T_{sf}) as shown in Eq. (6). Finally, the approximated optimal payload size is $\tilde{X_{opt}}$ in Eq. (7).

$$\frac{\partial max\left(T_{FH}\right)}{\partial X} = \frac{\partial max\left(T_{Q_R}\left(X\right) + T_{Q_G}\left(X\right)\right)}{\partial X}$$
$$= \frac{\partial\left(\frac{(min(X,E[x_{i,1}])+78)}{r/8} + \frac{\sum_{i=1}^{N}(E[x_i]+78\cdot M_i)}{R/8}\right)}{\partial X}$$
$$\approx \frac{\partial\left(\frac{(X+78)}{r/8} + \frac{\sum_{i=1}^{N}\left(E[x_i]+78\cdot \tilde{M}_i\right)}{R/8}\right)}{\partial X} \quad (5)$$
$$= \frac{1}{r/8} - \frac{78\cdot T}{R/8\cdot X^2} = 0$$

$$\frac{8}{T_{sf}}\cdot\sum_j\left(x_{i,j} + 78\cdot\left\lceil\frac{x_{i,j}}{X_{lb}}\right\rceil\right) = min\left(r_i, \frac{R}{N}\right), \forall i \quad (6)$$

$$\tilde{X}_{opt} = \begin{cases} max\left(\sqrt{\frac{78\cdot r\cdot T}{R}}, X_{lb}\right), & \text{If } E\left[x_{i,1}\right] \gg J \\ J, & \text{Otherwise} \end{cases} \quad (7)$$

Algorithm 1 summarizes the proposed packetization method in which the I_{pack} indicates these $x_{i,j}$ bits shall be packetized immediately or not stated in IV-A, X_b refers the remaining bits in the outgoing packetization buffer and $\tilde{X}_{opt}$ is the optimal maximum payload size in IV-B.

Algorithm 1 Packetization algorithm

Input: $x_{i,j}$ incoming bits with $I_{pack} = \{true, false\}$ (true only if these bits match conditions in IV-A);
X_b bits in the outgoing packetization buffer;
Output: Transmitted packet with X_b payload
while $x_{i,j} > 0$ **do**
  $X_{ori,b} \leftarrow X_b$
  $X_b \leftarrow X_b + min\left(x_{i,j}, \tilde{X}_{opt} - X_b\right)$
  $x_{i,j} \leftarrow x_{i,j} - (X_b - X_{ori,b})$
  if $X_b == \tilde{X}_{opt}$ **then**
    Send packet with payload equals to X_b
    $X_b \leftarrow 0$
  end if
end while
if $I_{pack} == true \wedge X_b > 0$ **then**
  Send packet with payload equals to X_b
  $X_b \leftarrow 0$
end if

V. Impact of Packet scheduling

In uplink case, all packets from different RRUs are scheduled and transmitted to the BBU pool over shared FH segment II. Moreover, in order to increase the FH efficiency, packets that are beyond the deadline can be discarded. These two schemes are applied at the RRU gateway.

A. Packet scheduling

A common scheduling policy is to allocate packets based on their arrival time; e.g., the first-come, first-served (FCFS). In this sense, different input queues can be viewed as a single FIFO queue. Before introducing other policies, we first formulate two useful metrics as follows.

- *Unscheduled bits of a symbol*: Unscheduled bits after the k^{th} packet (included) of j^{th} symbol from i^{th} RRU is $B_i\left(j,k\right)$ in Eq. (8) where the $P\left(i,j,k\right)$ is the size of the k^{th} packet at j^{th} symbol from i^{th} RRU in bytes.

$$B_i\left(j,k\right) = 8\cdot\sum_{l=k}^{n_i(j)} P\left(i,j,l\right) \quad (8)$$

- *Remaining time till deadline*: The remaining time till the deadline for packets of the j^{th} symbol from i^{th} RRU is $D_i\left(j,t\right)$ in Eq. (9) where $T_{symbol}\left(i,j\right)$ is the time after the acquisition of the j^{th} symbol from i^{th} RRU, $max\left(T_{proc}\right)$ denotes the upper bound of overall baseband processing time defined in Sec. III-D, and t is current time.

$$D_i\left(j,t\right) = T_{symbol}\left(i,j\right) + 2ms - max\left(T_{proc}\right) - t \quad (9)$$

Via utilizing the above two metrics, several packet scheduling policies can be applied at the RRU gateway are as follows:

1) **First-come, first-served (FCFS)**
2) **Shortest processing time (SPT)**: Select the RRU with the minimum packet size $P\left(i,j,k\right)$
3) **Least remaining bit (LRB)**: Select the RRU with the minimum $B_i\left(j,k\right)$. This policy prioritizes the RRU with the minimum remaining unscheduled bits.
4) **Earliest due date (EDD)**: Select the RRU with the packet that is closest to the deadline $D_i\left(j,t\right)$
5) **Least slack time (LST)**: Select the RRU with the minimum slack $S_i = D_i\left(j,t\right) - B_i\left(j,k\right)/R$. This policy further considers the remaining processing time compared with EDD.

B. Packet discard

When the slack of a packet is negative ($S_i < 0$), then at least one packets of the symbol cannot be delivered to the BBU on time. Therefore, this packet and all other packets of the same TTI from same RRU will be all discarded. This is because no further processing can be applied when the deadline expires. In addition, NACK will be sent from RRUs to all UEs on the downlink control channel in order to trigger the uplink data re-transmission by UEs. Besides, the same downlink data will be re-transmitted by the RRU on downlink data channel since no ACK is received in uplink direction.

VI. Simulation results

Numerical results and discussion of underlying insights on the maximum supported RRU number over a capacity-limited FH is presented. Most of the simulation parameters applied to UE, RRU, and BBU are from 3GPP standards (TS25.942, TS36.942, TS36.814) and NGMN documents [16]. For a fair comparison with [2], we use the 95^{th} percentile of the FH delay to decide the maximum supported RRU number.

A. Impact of packetization

In this paragraph, we provide the results consider the packetization impact with FCFS scheduling at RRU gateway.

1) Split A, B: Due to the high rate characteristic, the $\tilde{X}_{opt}$ equals to $X_{lb} = 5011$ and 2400 for Split A and B respectively. The FH delay with $\tilde{X}_{opt}$ is 5% and 9% less than the case using the jumbo frame payload size ($J = 8960$) displayed in Fig. 6 and Fig. 7. However, the supported RRU number in TABLE III is identical to the results in [2] due to the constant data rate characteristic of these two splits. In hence, no time-multiplexing benefit is observed.

TABLE III: Supported RRU number for split A, B

Split	Peak-rate analysis [2]	Packetization [2]	Packetization ($\tilde{X}_{opt}$)
A	5	5	5
B	9	9	9

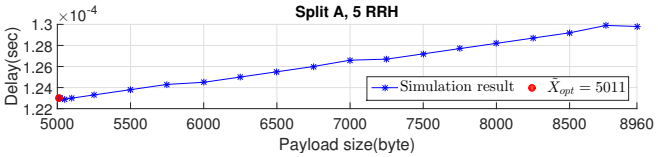

Fig. 6: Maximum payload size impact on split A

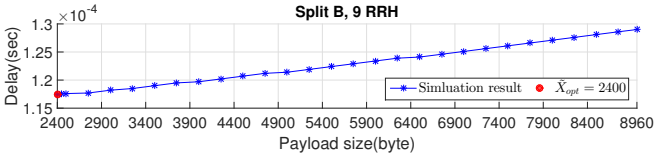

Fig. 7: Maximum payload size impact on split B

2) Split C: The impact of maximum payload size on the FH delay is in Fig. 8, and the approximated optimal payload size $\tilde{X}_{opt} = 4363$ which is very close to the local minimum point at around $[4250, 4500]$. The FH delay can be reduce by 12% after replacing the jumbo frame payload size with $\tilde{X}_{opt}$.

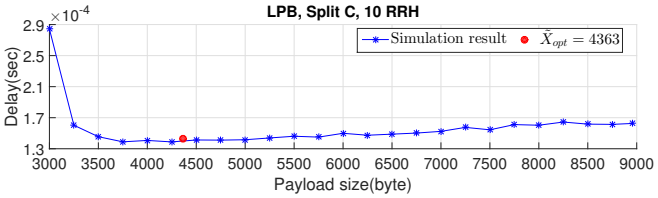

Fig. 8: Maximum payload size impact on split C

TABLE IV shows the supported RRU number compared with [2] under different UE-multiplexing bounds: Low Provisioning Bound (LPB), Conservative Lower Bound (CLB), Aggregated Common Bound (ACB) and Aggregated Strict Bound (ASB) [2], [17]. After considering the time-multiplexing impact and applying $\tilde{X}_{opt}$ as payload size, the number of supported RRU is increased.

TABLE IV: Supported RRU number for split C

Bound	Peak-rate analysis [2]	Packetization [2] (No multiplex)	Packetization (Multiplex and $\tilde{X}_{opt}$)
LPB	9	9	10
CLB	9	9	10
ACB	9	9	10
ASB	9	9	11

3) Split D: The maximum payload size impact on the FH delay is in Fig. 9 with six different channel estimation and demodulation operation modes (D1 to D6). Since the condition in Sec. IV-B can only be fulfilled by the D1 mode that with the largest look-ahead depth without pre-fetch scheme ($\tilde{X}_{opt} = 3741$); as a result, $\tilde{X}_{opt}$ for other modes (D2 to D6) equal to the jumbo frame payload size based on Eq. (7). Moreover, the FH delay of D5 and D6 mode is better than other modes since they apply less look-ahead depth and pre-fetch scheme.

TABLE V shows the supported RRU number compared with [2]. The operation mode of [2] is the same as D1 mode; however, the result is worse after considering time-multiplexing. This is because the average throughput in a subframe is used in [2] to get the over-estimated result without considering the bursty traffic impact at the end of each operation period. Moreover, the pre-fetch scheme brings 50% gain in the number of supported RRU when comparing the results of D1 and D2. It shows that the pre-fetch significantly reduces the instantaneous congestion. Further, as the look-ahead depth is decreased from 10 (D2) to 3 (D6), the supported RRU number increases due to bursty traffic mitigation. Nevertheless,

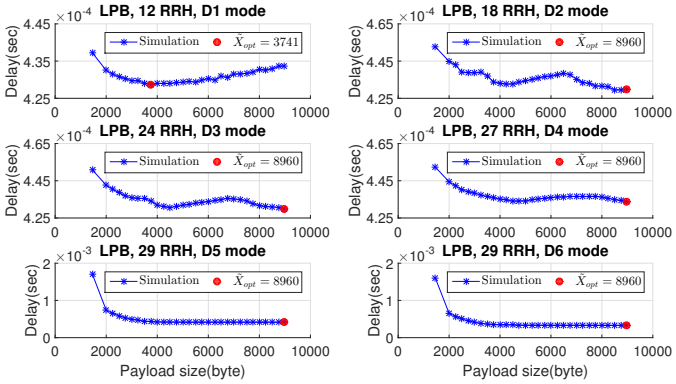

Fig. 9: Maximum payload size impact on split D

Operation mode	Operation period	Look-ahead depth	Pre-fetch
D1	subframe	10	No
D2	subframe	10	Yes
D3	subframe	6	Yes
D4	subframe	5	Yes
D5	subframe	4	Yes
D6	slot	3	Yes

using smaller look-ahead depth degrades channel estimation interpolation accuracy which is out of our scope in this work.

TABLE V: Supported RRU number for split D

Bound	Peak-rate analysis [2]	Packetization (No multiplex) [2]	Packetization (Multiplex and $\tilde{X}_{opt}$) Operation mode					
			D1	D2	D3	D4	D5	D6
LPB	7	18	12	18	24	27	29	29
CLB	7	16	11	17	24	27	29	29
ACB	7	13	12	18	22	24	26	26
ASB	7	14	13	19	25	27	29	29

4) Split E: The condition stated in Sec. IV-B cannot be fulfilled and $\tilde{X}_{opt}$ equals to the jumbo frame payload size. TABLE VI shows the supported RRU number compared with [2]. The result in [2] is also over-estimated since the average throughput in a subframe is used to derive the result without considering the bursty traffic impact. In contrast, the result provided in this work considers both the traffic characteristic as well as the time-multiplexing gain.

TABLE VI: Supported RRU number for split E

Bound	Peak-rate analysis [2]	Packetization [2] (No multiplex)	Packetization (Multiplex and $\tilde{X}_{opt}$)
LPB	66	215	153
CLB	66	213	154
ACB	66	208	141
ASB	66	160	142

To sum up, the maximum supported RRU number provided in this paragraph considers both the impact of packetization and the time-multiplexing gain. Further, several packetization schemes (Pre-fetch, look-ahead depth, Approximated optimal payload size) are proved to decrease the FH delay and increase supported RRU number.

B. *Impact of packet scheduling*

In this paragraph, only the result of Split E under LPB is shown for simplicity but the same trend happens on other splits and UE-multiplexing bounds. TABLE VII shows the number of supported RRU when applying different packet scheduling policies and packet discard scheme. It can be seen from TABLE VII that the LRB policy supports the most RRUs (162 RRUs) because it prioritizes the RRU with minimum remaining bits, i.e., the minimum-loaded RRU. Further, Fig. 10 shows the CDF plot of the FH delay when 162 connected RRUs,

and the LRB policy has the lowest 95^{th} percentile of FH delay. For the SPT policy, the number of supported RRU is a little larger than using FCFS since it prioritizes small packets without considering the load of RRU. In contrast, the LST policy prioritizes the RRU with the minimum slack, i.e., the minimum allowable processing time, so the FH delay from different RRUs are balanced. Accordingly, the variance of the FH delay is reduced shown in Fig.10, but the scheduling flexibility and the supported RRU number are decreased.

TABLE VII: Supported RRU number among schedule policies

Split	FCFS	SPT	LRB	EDD	LST
E	153	156	162	153	140

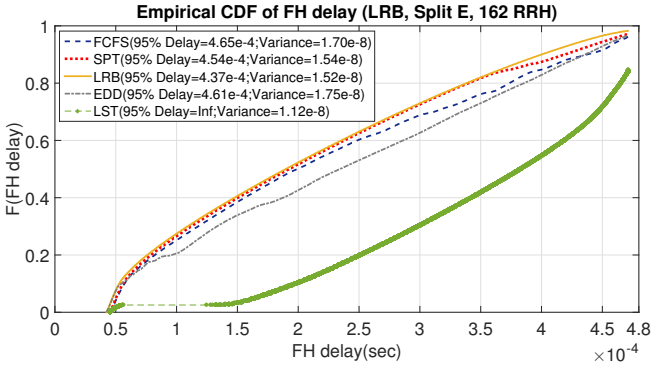

Fig. 10: CDF of FH delay for considered scheduling policies

The fairness in terms of the average FH delay and the average queue size is shown in Fig. 11. The LST policy has the highest fairness because it prioritizes the RRU with minimum allowable processing time, so the delay among RRUs are balanced. Moreover, the queue size is also fair for LST since the allowable processing time is in negative proportional to the number of unscheduled packets. However, the LRB policy has the worst fairness in the queue size due to it sets aside the packet from high-loaded RRUs. Fig. 12 shows the packet discard statistics considering discarded packet number and total discarded bits. The LST policy discards more packets but with fewer total bits; that is because fewer supported RRU number makes it discard more packets, but mainly small packets that with minimum slack are discarded. In contrast, the LPB majorly discards large packets from high-loaded RRUs, so the total discarded bits are large.

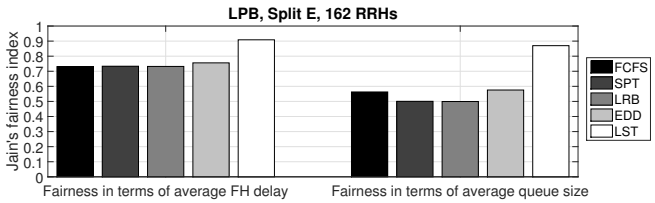

Fig. 11: Fairness in terms of average FH delay and queue size

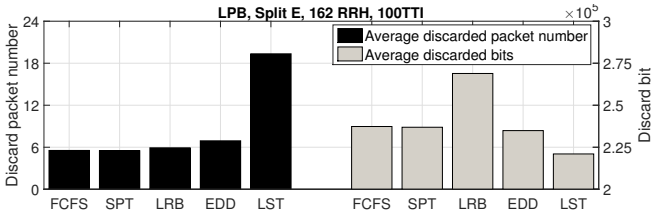

Fig. 12: Discarded packet characteristics

In summary, different scheduling policies show the trade-off between fairness and the multiplexing gain. The LST policy is a fair scheduler in terms of delay and queue size but provide less multiplexing gain, whereas the LRB policy performs well in terms of multiplexing gain but with less fairness. To this end, LRB showed to fit well for the considered FH capacity-limited problem.

VII. Conclusions and Future Work

This work focuses on packetization and packet scheduling for capacity-limited FH in C-RAN network to satisfy the HARQ-based timing constraint. Several schemes of packetization process are surveyed to decrease the FH delay and a packetization algorithm is proposed. The impact of different packet scheduling policies applied at the RRU gateway is also surveyed. Simulation results reveal that combining several packetization techniques (i.e., pre-fetch, short look-ahead depth and approximated maximum pay-load size) and the LRB scheduling policy with packet discard can significantly provide multiplexing gain in the maximum supported RRU number.

We are planning to further study the impact of processing job scheduling at RRU/BBU and analyze the joint capacity-limited and processor-limited problem.

Acknowledgments

Research and development leading to these results has received funding from the European Framework Program under H2020 grant agreement 671639 for the COHERENT project and from the French ANR Program under RADIS project.

References

[1] A. Checko *et al.*, "Cloud ran for mobile networks - a technology overview," *Communications Surveys & Tutorials, IEEE*, 2014.

[2] C.-Y. Chang *et al.*, "Impact of packetization and functional split on C-RAN fronthaul performance," in *IEEE ICC*, 2016.

[3] N. Nikaein, "Processing radio access network functions in the Cloud: Critical issues and modeling," in *MCSMS*, 2015.

[4] D. Wubben *et al.*, "Benefits and impact of cloud computing on 5g signal processing: Flexible centralization through cloud-ran," *Signal Processing Magazine, IEEE*, 2014.

[5] CPRI, "Interface specification v7.0," 2015.

[6] OBSAI, "Bts system reference document, v2.0," 2006.

[7] ETSI, "Open Radio equipment Interface (ORI); requirements for ORI," European Telecommunications Standards Institute(ETSI), GS ORI 001.

[8] IEEE, "1904.3 task force: Standard for radio over ethernet encapsulations and mapping," Tech. Rep., 2015.

[9] Y. Zhiling *et al.*, "White paper of next generation fronthaul interface (v1.0)," China Mobile Research Institute, Tech. Rep., 2015.

[10] I. Chih-Lin *et al.*, "Rethink fronthaul for soft ran," *Communications Magazine, IEEE*, vol. 53, no. 9, pp. 82–88, 2015.

[11] J. Bartelt *et al.*, "Fronthaul and backhaul requirements of flexibly centralized radio access networks," *Wireless Communications, IEEE*, 2015.

[12] U. Dötsch *et al.*, "Quantitative analysis of split base station processing and determination of advantageous architectures for lte," *Bell Labs Technical Journal*, 2013.

[13] K. Miyamoto *et al.*, "Analysis of mobile fronthaul bandwidth and wireless transmission performance in split-phy processing architecture," *Optics Express*, vol. 24, no. 2, pp. 1261–1268, 2016.

[14] I. Chih-Lin *et al.*, "Recent progress on c-ran centralization and cloudification," *Access, IEEE*, vol. 2, pp. 1030–1039, 2014.

[15] NGMN, "RAN evolution project backhaul and fronthaul evolution," NGMN Alliance, Tech. Rep., 2015.

[16] ——, "Next generation mobile networks radio access performance evaluation methodology," NGMN Alliance, Tech. Rep., 2008.

[17] ——, "Guidelines for lte backhaul traffic estimation (v0.4.2)," NGMN Alliance, Tech. Rep., 2011.