

Downlink Fronthaul Compression in Frequency Domain using OpenAirInterface

Cleverson Nahum, Leonardo Ramalho, Joary Fortuna, Chenguang Lu, Miguel Berg,
Igor Almeida and Aldebaro Klautau

Abstract—This paper presents a compression scheme developed for the transport of downlink radio signals in packet fronthaul of centralized-radio access networks (C-RAN). The technique is tailored to frequency-domain functional splits, in which inactive LTE (or 5G NR) resource elements may not be transmitted in the packets. This allows decreasing the link data rate, especially when the cell load is low. The compression scheme is based on two parts: transmission of side information to indicate active and inactive resource elements of the LTE, and nonuniform scalar quantization to compress the QAM symbols of the active resource elements. The method was implemented in the OpenAirInterface (OAI) software for real-time evaluation. The testbed results show a significant reduction in link usage with a low computational cost.

Keywords—OpenAirInterface, Fronthaul, Compression.

I. INTRODUCTION

The fifth-generation (5G) of mobile communications is expected to support a massive amount of users and provide very high data rates [1]. For that, it is required to increase system capacity, reduce communication latency, and increase reliability. Therefore, some features such as network densification, use of higher frequencies, and multiple antenna techniques are being explored [2]. To support the increased data rates, flexibility, new capabilities, and other demands of 5G, the Radio Access Network (RAN) is evolving. An example of this evolution is the Centralized RAN (C-RAN) architecture [3], which allows providing cost-effective mobile network implementation. In C-RAN an eNodeB is split into Remote Radio Unit (RRU) and centralized Baseband Unit (BBU), which communicate over a fronthaul (FH) link. BBUs are allocated in the data center, where it is possible to share computational resources amongst different sites providing more flexibility and costs saving. A drawback of C-RAN is the required high bit rate of the FH, which is the link used to transport the radio signal between BBU and RRU [4].

There are different approaches to decrease the FH rate in the literature, such as time-domain signal compression and

This work was supported in part by the Innovation Center, Ericsson Telecomunicações S.A., Brazil, CNPq and the Capes Foundation, Ministry of Education of Brazil, and by the European Union through the 5G-Crosshaul project (H2020-ICT-2014/671598).

C. Nahum, L. Ramalho, J. Fortuna and A. Klautau are with LASSE - 5G Group, Av Perimetral km 01, Guamá, 66075-750, Federal University of Pará, Belém, Brazil (e-mails: {cleversonnahum, leonardolr, joary, aldebaro}@ufpa.br).

Chenguang Lu and Miguel Berg are with Ericsson Research, 164 80, Stockholm, Sweden (e-mails: chenguang.lu,miguel.berg@ericsson.com,).

Igor Almeida is with Ericsson Research, Rod. Eng. Ermênio de Oliveira Penteado, 57,5 - Parque São Lourenço, Indaiatuba, Brazil (e-mail: igor.almeida@ericsson.com)

functional split [5]. For example, time-domain signal compression can be based on well-known methods such as vector quantization [6] or linear prediction and Huffman [7]. On the other hand, with functional split, the network processing stack is split between the BBU and RRU. Splits allow to decrease the FH bit rate, but the RRU performs more computational processing. Another solution is to combine both and use signal compression algorithms [8], [9] along with functional splits. An example of FH compression with the functional split is given in [10], where the authors propose a Baseband Fronthaul Termination (BFT) Nodes to make compression/decompression on FH and allocate resource blocks according to the cell usage reaching a fronthaul reduction of 33,3% in a full load system. One drawback of [10] is the huge increase of the computational complexity in the RRU because of a large part of the PHY layer is implemented in the RRU. A flexible functional split with A-law lossy compression for FH signals was presented in [11] reaching a fronthaul reduction of 50%. The Low-PHY split 7.1 was implemented in [11], which reduced the RRU computational cost but increased the FH bit rate because all subcarriers of an LTE symbol are sent, regardless of carrying or not useful data.

This paper presents an algorithm to compress and transport frequency domain representation of the LTE OFDM symbols. The proposed method is a combination of functional split and signal processing, where the RRU is responsible for two parts of the baseband OFDM signal generation: the IFFT and cyclic prefix insertion. The proposed algorithm, called *Bit Vector Compression* (BVC), discards the unused subcarriers before transporting data over the FH. Thus, the bit rate of the FH is proportional to the number of active subcarriers. After discarding the unused subcarriers, the IQ samples are quantized to reduce the FH bit rate further. In this paper, we used an A-law quantizer, but other scalar and vector quantizers can be used. We implemented the BVC in the Eurecom's Open Air Interface (OAI) software [12] to evaluate it in an online network. As in the original OAI's implementation, the LTE OFDM symbols are transported with the frequency domain representation over UDP packets in an Ethernet FH [11]. Since unused LTE Resource Elements (RE) are removed in this paper before the packet is sent through the FH, it is possible to reduce the FH data rate according to the data rate requested by the User Equipments (UEs). We show real-time results of data rate reduction and the relation between the FH and UEs' data rates.

Some of the contributions of the paper are listed in the sequel. The proposed algorithm has low complexity and can

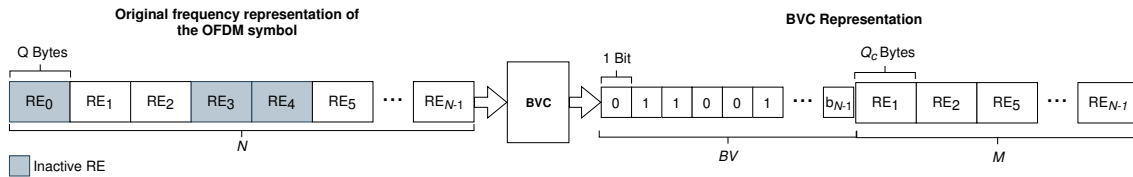


Fig. 1. Example of the bit vector compression (BVC) applied to a OFDM symbol, where each RE is as QAM symbol that modulates a subcarrier.

be used in scenarios of low-PHY functional split 7.1 [13] to send only used subcarriers from LTE symbols. The proposed compression method allows decreasing the FH rate according to the UEs' data rates. Furthermore, the proposed method could be implemented in low-cost RRUs, since the selected functional split implements few operations of the signal generation, and the proposed compression method is simple. Lastly, the results of the proposed algorithm are shown in a real testbed that can be modified in future works to implement other methods.

This paper is organized as follows. Section II presents a detailed description of the proposed compression method, explaining the algorithm and resources estimation accordingly to the number of active subcarriers. Section III shows the BVC implementation using an OAI testbed. Experimental results from the testbed are presented in Section IV. Section V concludes the paper.

II. PROPOSED METHOD

This section describes the proposed method to transport LTE signals more efficiently over a packet-based FH. For concreteness, we assume a 4G LTE network, but the method can also be applied to 5G New Radio (NR) scenarios. More specifically, we adopt OAI as a *reference implementation* of a packet-based LTE network scenario using 3GPP's 7.1 Low-PHY split [13], in which frequency domain data (also called subcarriers or QAM symbols) are sent through FH based on UDP transport. In the following paragraphs, we first present the foundations of the BVC, and then we provide equations to calculate the size of the packet that transports the compressed radio signal.

FH compression methods that work with the time-domain waveform (e.g., [6], [7]) send data corresponding to all REs of an OFDM symbol, regardless of the RE usage. In contrast, the proposed compression algorithm removes inactive REs from the packet of the downlink signals to achieve a bit rate that depends on the load in the wireless interface. It allows decreasing FH downlink traffic accordingly to the number of active subcarriers, which is proportional to the UEs data rates. In the BVC, one OFDM symbol is sent per UDP packet, where active REs and side information are sent. The side-information is called bit vector (BV) and indicates which subcarriers are active and inactive. Thus, the receiver can properly reconstruct the OFDM symbol in the frequency domain. The BV is a sequence of N bits that represents the activity in the N REs in the OFDM symbol, i.e., a value of 1 indicates active RE, and 0 indicates unused RE that is not transmitted over the FH.

The BVC is illustrated in Fig. 1, which also shows the BV field to indicate the active REs. On left side N REs of an original frequency representation of the OFDM symbol, where each RE is a complex QAM symbol that is represented with Q bytes. On the right side, all inactive REs were removed keeping M active REs, where each RE is quantized using A-law encoder becoming a representation with Q_c bytes. The bit vector BV is appended to the data to inform which subcarriers are being transported.

Fig. 2 shows the compression and decompression operations. Fig. 2(a) shows a flowchart of the operations that BBU performs before sending the samples to the RRU. The BVC maps the REs into the BV where each bit $BV[i]$ represents a RE of the OFDM symbol. The algorithm iterates over the REs verifying unused ones to assigns 0 to the respective BV position and removing the RE. For the used REs, the $BV[i]$ is assigned one to represent that the corresponding $RE[i]$ is active and the RE is encoded using A-law.

The RRU can reconstruct the original frequency domain representation of the OFDM symbol, after receiving the BV and active subcarriers data. The process is shown in Fig. 2(b), which represents the decompression algorithm. The BV and received REs are stored on BV and REs vectors. The algorithm iterates over each BV element verifying if it represents an inactive RE. If the current position in BV is one, then the next IQ sample of the RE array is decoded using A-law and inserted in the reconstructed array, and if the BV position is zero a null IQ sample (zero) is added in the reconstructed array. After all, REs were analyzed and the original data received is recovered to be used on RRU process.

Considering a FH using UDP transport, where each packet carries one OFDM symbol, so one LTE sub-frame is sent over 14 packets. The UDP packet size is dependent on the number of resources elements transported and the IQ symbol quantization. Using a fixed-length quantization scheme, all REs are represented with the same size. Besides LTE data, there is the header with information about LTE signal transported, Ethernet, IPv4, and UDP header. Therefore UDP packet size with no compression has a fixed value of

$$P_s = P_h + L_h + N \cdot Q, \quad (1)$$

where P_h represents the Ethernet, IPv4, and UDP header size. L_h represents LTE header with information about the OAI LTE, N is the number of resource elements and Q is the number of bytes used to represent IQ symbols. Typical values of P_h and L_h are 42 and 12 bytes, respectively [12]. Using the BVC, the BV field adds an additional overhead proportional to the number of REs in each UDP packet. Thus, UDP packet

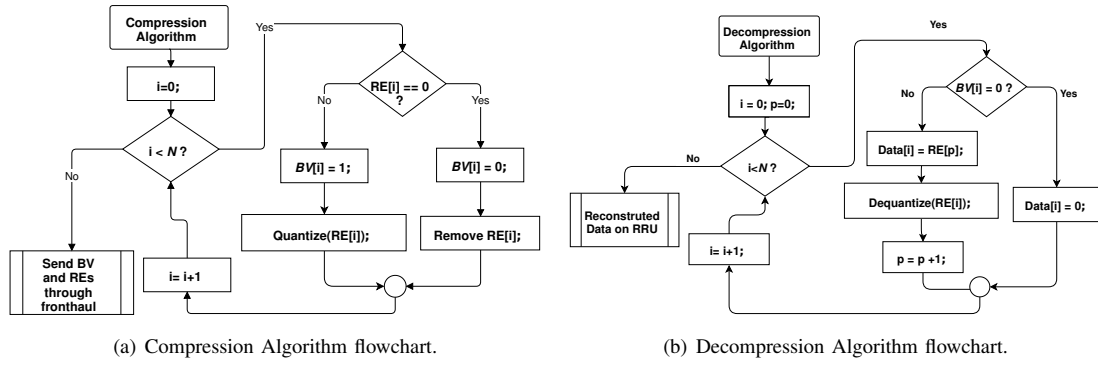


Fig. 2. Compression and decompression operations of the proposed BVC algorithm.

size using the BVC can be calculated as follows:

$$P_s = P_h + L_h + \left\lceil \frac{N}{8} \right\rceil + M \cdot Q_c, \quad (2)$$

where M represents the number of active resource elements and Q_c represents the number of bytes used in A-law encoder to quantize IQ symbols.

The compression efficiency varies according to IQ quantization and the number of REs being used. Considering a quantization of 2 bytes per RE and a 5 MHz bandwidth (maximum of 300 REs per OFDM symbol), when there are more than 281 active REs, the UDP packets are larger than when no compression is used because of the 38 bytes BV overhead added for 300 REs as perceived on Fig. 3.

Fig. 3 shows different packet sizes considering different cases using BVC and without BVC for a LTE signal of 5 MHz. These values can be calculated with (1) and (2). When BVC is used, the packet size is evaluated with $Q_c = 1$ and $Q_c = 2$ bytes per complex sample. Without BVC, the number of bytes per complex sample is also one and two. When there is no active RE, the BVC packets have just headers and the bit vector information. For the cases using BVC, when the packet size is larger than without method, an additional field could be used to indicate whether the BV field is present or not. Thus, it could be a way of enabling or disabling online the BVC to deal with its overhead disadvantage.

III. VALIDATION ON OAI-BASED TESTBED

A. OAI implementation

The BVC was implemented by modifying the OAI software, which is composed of two main source code repositories: 'openairinterface5g' (OAI5G) and 'openair-cn'. The project openair-cn has the source-code implementation of the evolved packet core (EPC) network. OAI5G repository contains source-code implementation for RAN LTE stack, it can operate as a complete eNodeB or as BBU and RRU, and also implement UE. It is also possible to make a functional prototype using Software Defined Radio (SDR) acting as RRU.

Fig. 4 (a) shows the LTE functional split 7.1 [5] defined by 3GPP, which is called in OAI project as IF4.5 split. In this scenario, RRU implements IFFT, cyclic prefix, parallel to Serial (P/S) conversion and RF transmission. Fig. 4 (b)

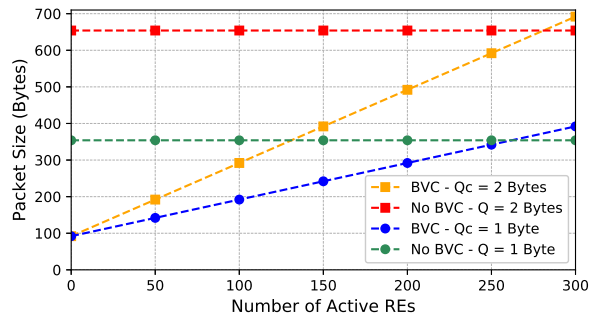


Fig. 3. Variation of the packet size with the number of bytes per IQ symbol and number of active REs.

illustrates in which machine OAI modules are executed and their connections. BBU and RRU are connected through an Ethernet switch, and the packets are UDP besides background traffic. The Ethernet link is shared with other services being susceptible to network congestion, delay and packet loss which are handled by OAI and minimized using the BVC. A USRP B210 is used as an RF transceiver, and it is connected to the RRU machine through a USB 3.0.

Fig. 5 illustrates OAI operations and where the BVC was implemented. In BBU's process called resource mapping, information is represented with 4 bytes for each complex-valued RE (2 bytes to the real part and 2 bytes to imaginary part). BVC removes inactive REs and performs A-law encoding to the active REs. Then, the information is sent through a UDP socket and received on RRU, where all inverse operations are applied, and the original frequency representation is reconstructed. In summary, the OAI software was altered to include the bit vector compression and bit vector decompression, in the BBU and RRU, respectively.

OAI sends one OFDM symbol per UDP packet, so considering a 5 MHz scenario without compression, it can send 300 REs per packet. Each packet has a header of $P_h = 42$ bytes (UDP, IPv4 and Ethernet headers) plus $L_h = 12$ bytes of IF4.5 header added to control frame information and each RE is quantized with $Q = 2$ bytes provided by A-law encoder. So when BVC is not used, to send zero active REs each packet has 654 bytes, as indicated in Fig. 3 because it always sends

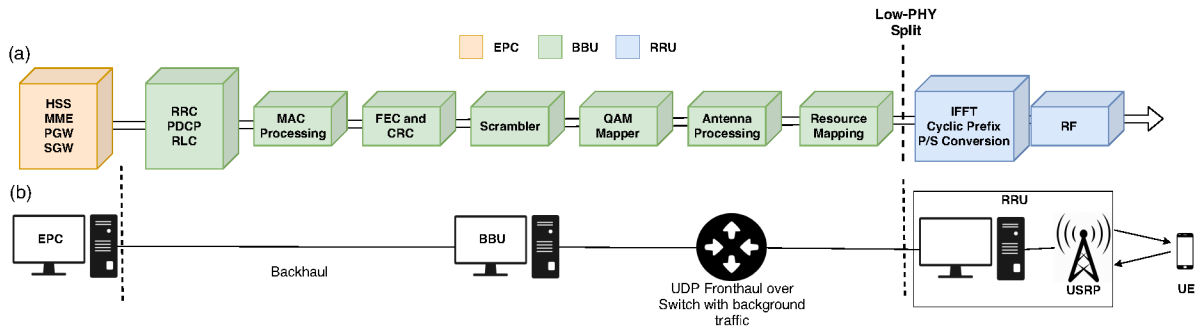


Fig. 4. (a) Split 7.1 (Low-PHY) of 3GPP LTE Functional split, which is called in OAI as IF4.5. (b) Testbed machines where OAI modules are working and USRP used to transmit RF signals to UE (smartphone)

information, independently if REs are used or not. On the other hand, when BVC is used, packets with 0 active REs have only 92 bytes that correspond to header and bit vector information. This value varies according to the number of active REs.

B. Hardware

The BBU machine uses a Core i7-5930K@3.50 GHz processor with 16 GB RAM and RRU machine uses a Core i5-4590@3.30 GHz with 8 GB RAM, both are using kernel 3.19.0-61-lowlatency (real-time operations) and Ubuntu 14.04. EPC runs into a virtual machine with customized kernel 4.7.7-oaiepc made by OAI inside BBU machine and using Ubuntu 16.10, sharing resources and with 4 GB RAM allocated. The OAI was configured to run in Frequency Division Duplex (FDD) with a central frequency of 2685 MHz. The LTE signal has 5 MHz bandwidth, which corresponds to a maximum of 25 Resource Blocks (or 300 subcarriers). The fronthaul was implemented with a 1 Gbps Ethernet connection among machines with 1400 bytes for MTU size.

A USRP B210 was connected to the RRU machine to establish radio access to OAI network. A Samsung Galaxy S4 smartphone was used as UE connecting to RRU through USRP and generate user plane traffic over FH. Iperf tool was used to create traffic to UE through OAI network to test and obtain results.

IV. EXPERIMENTAL RESULTS

The experimental results were captured with the iperf tools, where it was used an iperf client on UE and an iperf server on EPC machine to generate traffic over the FH downlink. Fig. 6 shows UE download bit rate and FH downlink throughput with and without BVC.

Without BVC, the FH traffic is constant, but when the proposed method is applied the FH throughput changes with the UE downlink bit rate. As indicated before, the BVC algorithm compresses the data according to UE download rate. As shown in Fig. 6, there is a reduction of 75.75% on BVC FH throughput in relation to FH throughput without BVC when there is only UE control data. Comparing with the FH bitrate necessary to transport time-domain samples presented in [11], the reduction of the FH bitrate is approximately 92%.

The BVC FH throughput grows proportionally when the iperf application starts to require data on the UE. Therefore, with only control data, the fronthaul usage is about 24.25% of the total FH capacity, the growth is about 4.5% of total FH capacity per 1 MB required by UE.

Actually, there is a linear relation between the UE data rate and the FH bitrate, as shown in Fig. 7. When UE download rate is zero, there is only control traffic being transported over FH (control channels and references signals of LTE). After UE starts to require bandwidth, the control traffic is combined with UE data, which increases FH throughput proportionally to the number of REs being requested by UE. Therefore, the more bandwidth is requested by UE, more REs are allocated to transport data, and consequently more FH rate is needed.

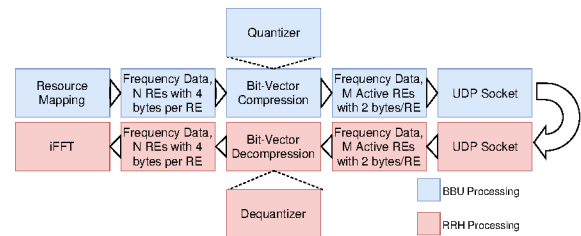


Fig. 5. Implementation of the bit vector compression and decompression on the BBU and RRU for downlink signals.

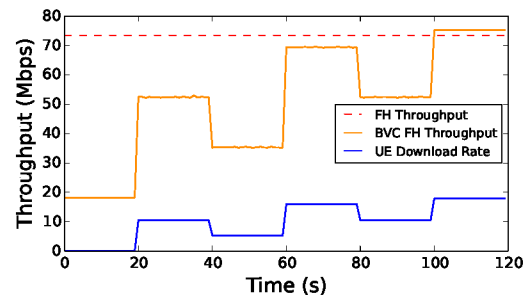


Fig. 6. UE download speed and FH downlink throughput.

The BVC was also evaluated in terms of processing time. Then, the results were compared with the method shown

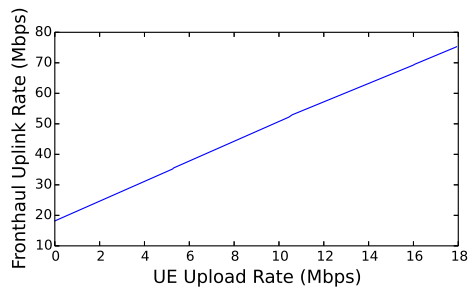


Fig. 7. Relationship between the UE and FH throughput.

in [11], where A-law is also used to reduce FH traffic. Compression and decompression operations time were collected to measure the processing impact of using the BVC technique. Considering a 5 MHz scenario (300 REs), it was measured time to execute bit vector mapping and A-law quantization. Figures 8 and 9 show boxplots of the execution time for compression and decompression, respectively. Different values of active REs were used, where the results of [11] are represented by the case with 300 active REs.

Fig. 8 shows the time to compress a certain number of active REs. It would be expected longer execution time due to the mapping process added by BVC. However, analyzing the compression time, it was perceived that there is a trade-off because when a few REs are sent, a few REs will need to be quantized, which reduces the processing time comparing to the results of [11](300 REs). It is perceived that up to 90 REs being used, BVC presents a lower processing time despite the mapping process. With more than 238 REs, time tends to be longer, but this difference is less than 150 nanoseconds for 238 REs case.

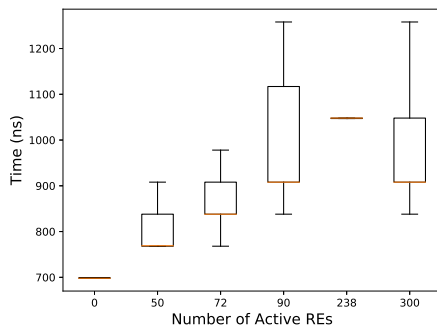


Fig. 8. Time required by compression operations.

The decompression time is shown in Fig.9. In this case, the decompression time for the BVC algorithm is always longer than [11](300 REs). The results show this behavior because BVC always need to verify all REs. Therefore, there is no processing gain using the BVC to decompress LTE symbols on RRU. But processing time overhead is less than 0.3 microseconds, and the OAI operations were not affected by this overhead. Moreover, the FH bit rate can be considerably reduced.

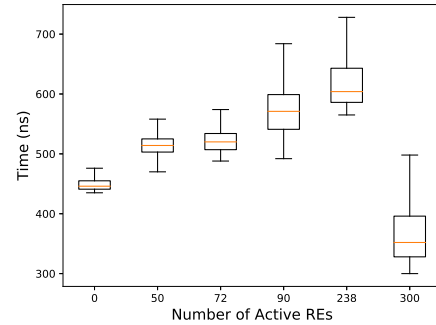


Fig. 9. Time required by decompression operations.

V. CONCLUSION

This work has proposed a bit vector mapping method combined with A-law quantization applied to REs in a packet-based FH. The proposed method works in a Low-PHY functional split that transports the frequency domain representation of the signals. The proposed algorithm was implemented and tested in OAI platform where FH traffic is reduced according to cell load, reaching 75% reducing when there is only control traffic for frequency domain, and 92% of FH bitrate reduction when compared with the transportation of the time domain signal. The proposed BVC has presented a low processing overhead in relation to processing using just A-law quantization, offering a better FH usage. In the future, BVC can be extended to support uplink.

REFERENCES

- [1] E. Dahlman *et al.*, "5G wireless access: requirements and realization," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 42–47, 2014.
- [2] S. Chen and J. Zhao, "The requirements, challenges, and technologies for 5G of terrestrial mobile telecommunication," *IEEE communications magazine*, vol. 52, no. 5, pp. 36–43, 2014.
- [3] I. Chih-Lin *et al.*, "Recent progress on C-RAN centralization and cloudification," *IEEE Access*, vol. 2, pp. 1030–1039, 2014.
- [4] J. Bartelt *et al.*, "Fronthaul and backhaul requirements of flexibly centralized radio access networks," *IEEE Wireless Communications*, vol. 22, no. 5, pp. 105–111, 2015.
- [5] L. M. Larsen *et al.*, "A survey of the functional splits proposed for 5G mobile crosshaul networks," *IEEE Communications Surveys & Tutorials*, 2018.
- [6] H. Si *et al.*, "A novel and efficient vector quantization based CPRI compression algorithm," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 7061–7071, Aug 2017.
- [7] L. Ramalho *et al.*, "An LPC-based fronthaul compression scheme," *IEEE Communications Letters*, vol. 21, no. 2, pp. 318–321, 2017.
- [8] S.-H. Park *et al.*, "Fronthaul compression for cloud radio access networks: Signal processing advances inspired by network information theory," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 69–79, 2014.
- [9] S. Pawar *et al.*, "Front-haul compression using scheduling side information for cloud radio access networks," in *2015 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–6.
- [10] J. Lorca and L. Cucala, "Lossless compression technique for the fronthaul of LTE/LTE-advanced cloud-RAN architectures," in *2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*. IEEE, 2013, pp. 1–9.
- [11] C.-Y. Chang *et al.*, "FlexCRAN: A flexible functional split framework over ethernet fronthaul in Cloud-RAN," in *2017 IEEE International Conference on Communications (ICC)*. IEEE, 2017, pp. 1–7.
- [12] "OpenAirInterface – 5G software alliance for democratising wireless innovation." [Online]. Available: <http://www.openairinterface.org/>
- [13] G. T. 38.801, "Study on new radio access technology: Radio access architecture and interfaces," 2017.