Understanding Energy Consumption of Cloud Radio Access Networks: an Experimental Study

Ujjwal Pawar, Aditya Kumar Singh, Keval Malde, Bheemarjuna Reddy Tamma, and Antony Franklin A
Indian Institute of Technology Hyderabad, India
{cs19mtech01006, cs19mtech11032, cs20mtech01003, tbr, and antony.franklin}@iith.ac.in

***Abstract*—Cloud Radio Access Network (C-RAN) is rising as an attractive solution for the operators to cope with the ever-increasing user demand in a cost-efficient way. C-RAN's architecture consists of (i) Distributed Units (DU) located at the remote sites along with RF processing units, (ii) the Central Unit (CU) consisting of high speed programmable processors performing tasks such as mobility control, radio access network sharing, positioning, session management over a (iii) low latency, high bandwidth fronthaul link, which connects multiple DUs to the CU pool realized on a cloud platform. In traditional C-RAN, the functionalities that the BBUs and RRHs have to perform are fixed. Instead of having such a fixed set of functionalities, the concept of functional splits was introduced by 3GPP to bring forth the idea of shifting network stack functions between CUs and DUs in next generation C-RAN. In this paper, a real-time C-RAN testbed running on OpenAirInterface (OAI) software platform is used to profile the energy consumed by different functional splits configured by varying the CPU clock frequency and channel bandwidth. It is observed that for some lower CPU clock frequencies, the energy consumption is reduced without affecting the system throughput and overall user experience. With these insights, operators can improve the energy efficiency of C-RAN systems deployed.**

***Index Terms*—C-RAN, OpenAirInterface (OAI), Functional Splits, and Energy Profiling.**

I. Introduction

The main reason for migration from LTE to 5G is propelled by the increase in the number of users, diverse use cases, and their demand for high-speed connectivity. This imposes a huge burden on telecom operators as CAPEX and OPEX [1] go up for deploying new infrastructure with increased cell density. 5G networks will be much more efficient on a per-bit basis. However, they are set to carry much more user traffic over more cell sites powered by energy-hungry Massive MIMO antennas. Hence, the operators could face up to 2-3 times higher energy costs compared to 4G. To address this challenge, a novel network architecture called Cloud or Centralized Radio Access Network (C-RAN) [2] is proposed for the next generation cellular networks. C-RAN base station consists of two main components: one being Distributed Unit (DU) for carrying out low-PHY and RF functionalities and the other being Central Unit (CU) for carrying out the rest of stack functionalities over a low latency, high bandwidth fronthaul link. Of its vast benefits promised, some include the ability to pool resources, virtualize real-time systems, support multiple technologies, reduce energy consumption, reuse and simplify infrastructure thereby reducing CAPEX and OPEX. The CU-DU network protocol stack as proposed by the 3rd

978-1-7281-7299-6/20/$31.00 © 2020 IEEE

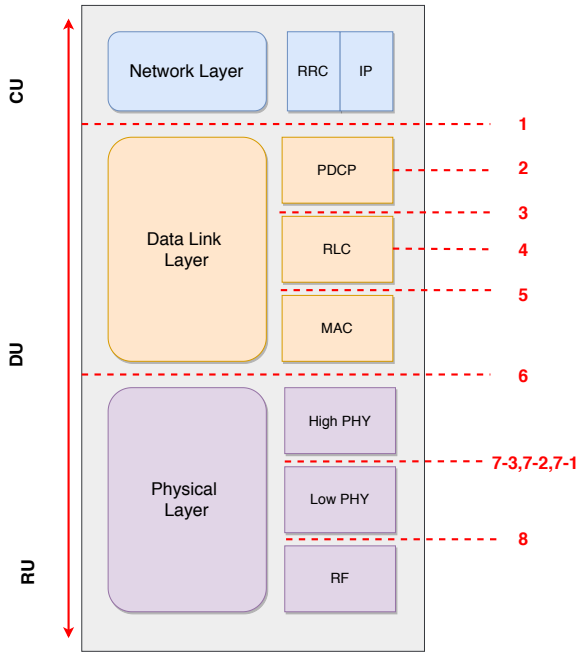

Fig. 1. The LTE protocol stack with layers and sublayers, including the numbered functional split options proposed by 3GPP

Generation Partnership Project (3GPP) is shown in Fig. 1. The numbers running from 1 to 8 are called functional splits wherein each split option characterizes the functionalities at the CU and DU/RU. For instance, the split 8 option represents the configuration where all the functionalities are present at the CU and only the RF unit is present at cell site. Therefore, the higher the split number, the more processing has been moved to the CU pool which is realized on a cloud platform setup using commodity servers.

There have been a number of papers addressing C-RAN's benefits and performance for different split options theoretically. Out of the few papers which showed practical results, [3] addressed the performance in terms of CPU and memory usage for splits 7 and 8. In [4], CPU utilization was measured by varying number of PRBs and MCS; and LTE sub-frame processing time was calculated by varying CPU frequency

without considering various split options. In [5], CPU utilization by different functions of OAI platform was observed by varying the type of data transmission.

In this paper, for a higher level split and a lower level split, the energy consumption and effective CPU utilization by varying CPU frequencies and channel bandwidth (i.e., Physical Resource Blocks (PRBs)) without any decrease in system throughput and User Experience are presented. In conjunction with that, an ideal frequency and minimum number of CPU cores required in the underlying cloud platform to run these split options are identified.

II. System Model

This section briefly describes C-RAN system architecture and the concept of functional splits.

A. C-RAN: System Architecture

The traditional C-RAN architecture consists of Baseband Units (BBUs) which are responsible for processing and control, and Remote Radio Heads (RRHs) which handle the radio transmission of the processed signal. The drawback of this architecture is that the number of BBUs is to be increased if the traffic and user demands are increased. And also, due to the heterogeneity of the traffic, a majority of BBUs might be under-utilized most of the times in a day. This will cause additional cost of deployment and maintenance, and low energy efficiency for the operators. Hence, traditional C-RAN architecture is not suitable for 5G wherein we expect to see a massive deployment of cells with high channel bandwidths. Therefore, with the main aim of centralizing and virtualizing the functionalities of highly-demanding 5G networks, next generation C-RAN architecture with various functional splits was proposed by 3GPP. The architecture consists of Radio Units (RUs) at the cell sites, Distributed Units (DUs) near cell sites at edge cloud or in the CU pool or at cell site with RUs, and Central Units (CUs) in a regional cloud as shown in Fig. 2. Note that, RUs mostly consist of RF and a few functionalities of LOW-PHY layer depending on the split configuration. These components are interconnected using a low-latency high bandwidth fronthaul interfaces supported by various technologies like e-CPRI.

B. Functional Splits

In traditional C-RAN, the functionalities that BBUs and RRHs have to perform are fixed. As described above, there are multitude of disadvantages with such an architecture. Therefore, instead of having such a fixed set of functionalities, the concept of functional splits was introduced to bring forth the idea of shifting network stack functions between CUs and DUs in the next generation C-RAN. Based on the LTE protocol stack, which 5G adopted with minimal changes, the 3GPP proposed 8 such functional splits. As seen in Fig. 1, higher numbered splits imply that a majority of processing of PHY and Data link layers has been done at the CU pool. Based on traffic demands, split configuration and channel bandwidth

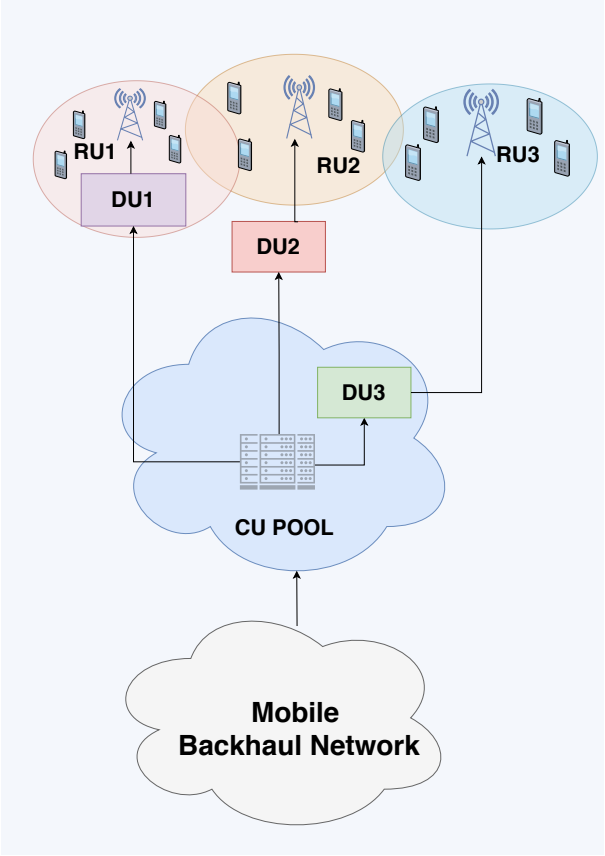

Fig. 2. An Example of C-RAN Architecture

can be adjusted by the operators for reaping in energy saving benefits.

Out of the lower layer splits, option 8, belongs to the traditional C-RAN architecture where only the RF unit is left at the DU (also known to be RU) and the rest of the layers of the stack are centralized at a CU pool. Since most of the processing has been done at the CU, it is easy to have a small and cost effective RUs but with stricter latency and high fronthaul bandwidth requirements. This option can act as an enabler for technologies such as CoMP and multi-RAT systems. The split option 7 is divided into three sub-options specified as 7-1, 7-2, and 7-3 that further divide the functionalities of the PHY layer. For all these three sub-options also, high bandwidth and low-latency requirements are imposed. Even though CoMP schemes are possible if CU/DU of multiple cells are co-located and jointly processed, it would still require a complex timing synchronization between the RU and CU/DU links of multiple cells. Split option 6 separates PHY and Data link layer resulting in a significant reduction in the fronthaul bandwidth requirements as compared to the higher numbered splits discussed so far. The main advantages include joint transmission, centralized scheduling, and resource pooling. But its drawbacks include the effect on HARQ timing and scheduling due to round trip fronthaul delay and the subframe-level timing interactions between the MAC in the CU and

the PHY in the DUs. Split 5 divides the MAC into High and Low MAC layers to reduce the latency requirements as HARQ processing and cell-specific MAC functions are performed at the DU itself. But the problem of complex CU-DU interface and the difficulty in defining scheduling operations exist even if there are low bandwidth requirements. Split 4 separates the MAC at the DU and the RLC at the CU. Low fronthaul bandwidth requirements are expected here and the bit rates scaled with MIMO layers configured. Split 3 places the PDCP and High-RLC layers at the CU and the rest of layers at the DU. Again here, low-latency and low bandwidth requirements are expected. In its sub-options, split 3-1 is more latency-sensitive than 3-2 due to ARQ being in the CU and not in the DU. Split option 2 enables centralization of PDCP layer of multiple cell sites. Thus, it is suitable for high layer split between CU and DU, and also tolerates high latency. Split option 1 places the entire Data link and PHY layers at the DU making it the complex version of C-RAN system deployed at cell sites but with very low bandwidth and relaxed latency constraints on the fronthaul.

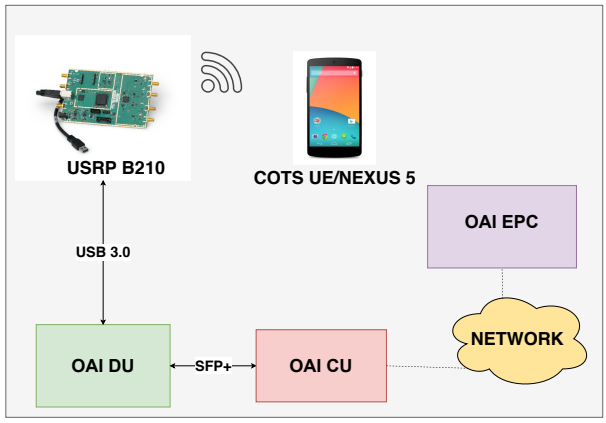

Fig. 3. Illustration of C-RAN testbed architecture using OpenAirInterface

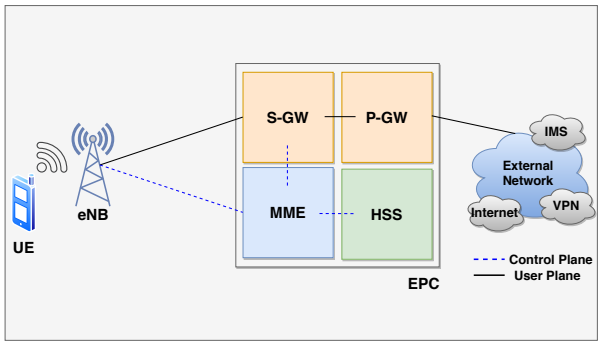

Fig. 4. Evolved Packet Core (EPC) Architecture

III. Experimental Setup

This section describes the experimental setup and the tools used to perform energy profiling of C-RAN split options.

A. Testbed Description

Fig. 3 illustrates the testbed setup used for experimentation which is based on OpenAirInterface (OAI) platform [7]. OAI is an open source software implementation of 3GPP cellular network architectures with support for C-RAN. So far, it supports three split options i.e., options 2, 7, and 8. In our experiments, split options 2 and 7 are used. For Evolved Packet Core (EPC), OpenAir-CN [7] component is used (refer Fig. 4). EPC consists of the following network elements: Mobility Management Entity (MME), Home Subscriber Server (HSS), Serving Gateway (S-GW), and PDN Gateway (P-GW). The RF part is realized using Ettus USRP B210 SDR. Ubuntu 18.04 is used to run both DU and CU parts on Intel® 6 Physical core (12 logical core) Xeon W series skylake based processor with 140 Watt TDP, clocked at 3.6 GHz and 16 GB DDR4 RAM. This CPU architecture supports Intel® AVX2 instructions. For User Equipment (UE), a Commercial Of The Shelf (COTS) UE is used with programmable SIM card.

B. Monitoring Tools

In the following, various performance monitoring tools used for energy profiling of C-RAN functional splits are described in brief.

1) Perf: Perf [8] is a performance analyzing tool for Linux based systems that abstracts away underlying CPU hardware differences. It is used to capture various events like CPU utilization, effective cycle, IPC, etc. CPU utilization metric gives the time the CPU was not running the idle thread. It also calculates migrations that capture the essence of moving a virtual CPU from one run queue to another as per scheduler.

2) PCM Tool: Processor Counter Monitor (PCM) [10] is an application programming interface (API) and a set of tools based on the API to monitor performance and energy metrics of Intel processors. PCM has a basic processor monitoring utility which can report instructions per cycle, core frequency, local and remote memory bandwidth, cache misses, core and CPU package sleep C-state residency, core and CPU package thermal headroom, cache utilization, CPU and memory energy consumption.

3) Intel Vtune: Vtune [9] is a software tool developed by Intel® which can be used to analyze, tune, and optimize the performance metrics in systems. Its capabilities include optimizing single and multi-threaded use of cores, providing a system level overview of application performance, diagnosing memory, storage, and data plane bottlenecks. It supports multiple programming languages, compilers, host and target operating systems.

IV. Experimental Results

In this section, energy consumption, CPU utilization, CPU migrations, system throughput, front-end bound, and back-end bound results are delineated. These results are collected on

the OAI based C-RAN testbed by using various monitoring tools presented in the above section. To observe the effect of energy consumption at different CPU clock frequencies, various tests are conducted on C-RAN testbed; one of them is to obtain the energy results using the PCM tool. As the PCM tool gives system-wide energy and DRAM energy, the energy consumed is calculated as the difference between the energy value reported when the system was in idle state and the energy value reported when the tests were running on the system.

A. Variation in energy consumption in different C-RAN functional split options

The main objective of this experiment is to compare the energy consumption of a higher layer split option to a lower layer split option. Out of the available split options, options 2 and 7 are considered as higher layer functional split and lower layer functional split, respectively. The CPU frequency is varied from 3.6 GHz to 1.5 GHz by keeping the channel bandwidth as 5MHz (or 25 PRBs).

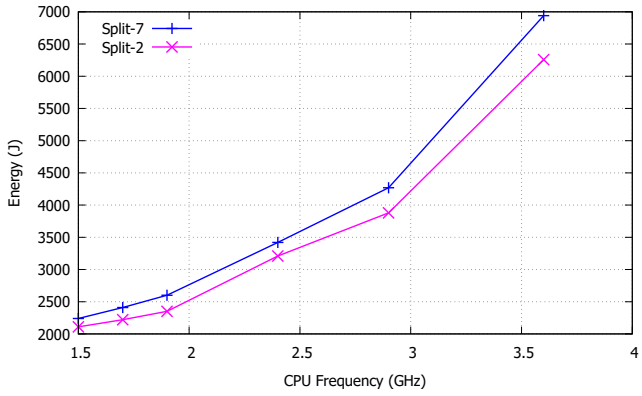

Fig. 5. CU energy consumption vs CPU frequency for split options 2 and 7

The user traffic is generated in full-buffer mode using iperf3 for 120 seconds in uplink direction. Fig. 5 shows the energy consumption at CUs in both split options for various CPU frequencies. The CU in split option 2 contains only RRC and PDCP layers and therefore consumes lesser energy when compared to the CU in split option 7 which in addition to RRC and PDCP also implements RLC and MAC layers. Fig. 6 shows the energy consumption at DUs in both the split options. The DU of split option 2 consists of RLC, MAC, and PHY leading to higher energy consumption as compared to DU of split option 7 as it contains only PHY layer. Fig. 7 shows the total energy consumption (i.e., sum of both CU and DU energies) which turns out to be almost the same for both the split options as this study excludes energy consumed by fronthaul links.

B. Effective CPU Utilization Estimation

To estimate the effective CPU utilization, various experiments are conducted. In the first set of experiments, all the 12 logical cores of the system are kept active and results are

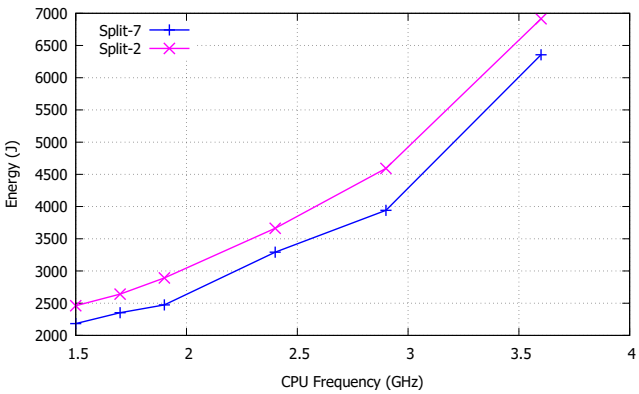

Fig. 6. DU energy consumption vs CPU frequency for split options 2 and 7

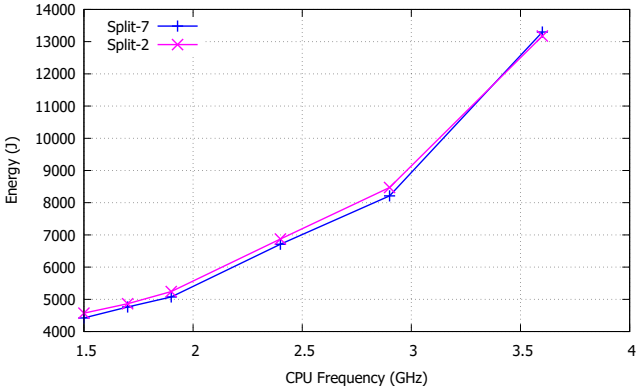

Fig. 7. Total energy consumption vs CPU frequency for split options 2 & 7

collected using *vtune* tool. After this, to study the effect of active CPU cores, a few CPU cores are turned off and similar results are collected; but this time, using *Perf* tool as *vtune* does not work when some of the cores are made inactive. The *Perf* tool derives the effective CPU utilization from the ratio of task_clock for a particular process and its total execution time.

1) Effective CPU utilization for Split Options 2 and 7: The objective of this experiment is to compare the effective CPU utilization of a higher layer split option to a lower layer split option when all cores are kept active. In this experiment, user traffic is generated using iperf3 for 120 seconds in both uplink and downlink directions. Fig. 8 shows the effective percentage CPU utilization of CUs of both split options 2 and 7 for uplink iperf traffic. The CU in split option 2 is very light weight containing only RRC and PDCP layers and hence having very low CPU utilization as compared to that of split option 7. On the contrary, Fig. 9 shows percentage CPU utilization of DUs. Here, DU of split option 7 has lower CPU utilization compared to that of split option 2 as it only contains low PHY and RF components. But this utilization is still significant and it is mostly from Linux system calls for the functioning of the RF part. At 1.7 GHz CPU frequency, for split option 2 the throughput is reduced at the UE and hence

it is not included in the plot. It can also be noticed from the plots that effective CPU utilization decreases with increase in CPU clock frequency as the processing could be accomplished easily by more powerful CPU cores when we increase their clock frequencies.

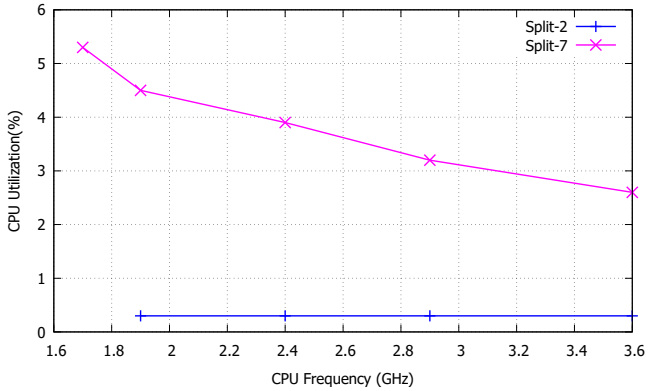

Fig. 8. CPU utilization of CU for splits 2 and 7 for uplink traffic

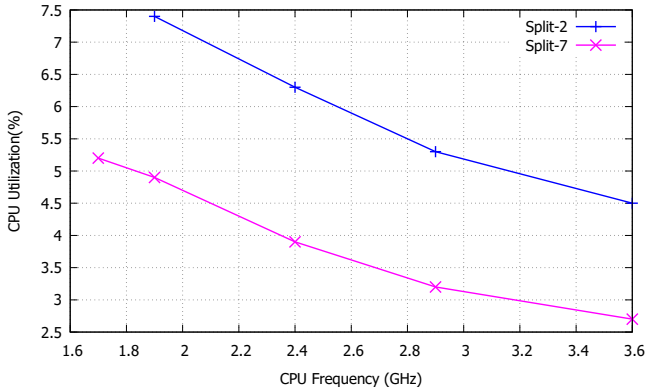

Fig. 9. CPU utilization of DU for splits 2 and 7 for uplink traffic

2) CPU utilization at different bandwidths: To understand the computational requirements at different channel bandwidths (or PRBs), experiments are conducted at 5MHz (or 25 PRBs) and 10 MHz (or 50 PRBs) for split option 2. User traffic is generated using iperf3 again in both uplink and downlink directions for 120 seconds. As the CU of split option 2 has very less processing requirement, it hardly changes when moved from 25 PRBs to 50 PRBs. In the idle condition, only UE is connected and no user data is being transmitted.

Fig. 10 shows the variation in percentage CPU utilization of DUs with channel bandwidths. It contains CPU usage in idle condition for both 25 PRBs and 50 PRBs. In the 50 PRBs case, an increase in CPU utilization is observed because most of the processing is carried out by DU in split 2 case. The increase is not double but is still significant. It has to be noted that for 50 PRBs, from 2.4 GHz onwards the user throughput is decreasing as the system is unable to process the incoming traffic in time. At 1.5 GHz for 50 PRBs, the system is unable to even decode anything and therefore zero throughput is reported by the UE. Due to this reason those frequencies are not included in the figure. Whereas for 25 PRBs from 1.7 GHz on wards, the user throughput is decreasing and at 1.5 GHz the user throughput is reduced to 2-3 Mbps but does not become zero. Similar experiments are conducted in the downlink direction and the results show similar trends.

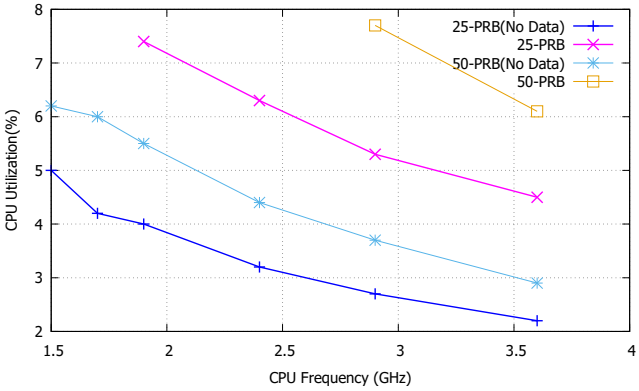

Fig. 10. CPU utilization for split 2 at different bandwidths with uplink traffic

3) Effective CPU utilization on 2 active logical CPU cores: The effect of reducing number of active cores and varying CPU frequency on system throughput and overall user experience was studied in this experiment. The test setup consists of three systems running EPC, CU, DU, respectively, all on Intel x86 along with OAI-CN and OAI-RAN packages. The experiments are conducted for functional split 2 DU and functional split 7 DU. CU is kept at 1.5 GHz frequency and only two logical cores are used throughout the experiment. As the number of cores is fixed to 2 at DU and frequency is varied from 1.5 GHz to 3.6 GHz, this setup has fixed number of 25 PRBs in both uplink and downlink directions. Perf calculates effective utilization based on ”non-ideal time” i.e., time the CPU was not running the idle thread.

Fig. 11 shows the effective CPU utilization with change in CPU frequency at different frequency levels. In the case of split option 2 at 1.5 GHz frequency, it is observed that effective utilization is 0.83, but for the same 1.5 GHz the option 7 used less than 1 core effectively. At 3.6 GHz for both split options 2 and 7, it was under 0.5 i.e., not even half of a core is effectively utilized.

Fig. 12 shows the same results for uplink data traffic between split option 2 and split option 7, at 1.5 GHz. Split 2 was able to consume entire 1 core effectively. At 3.6 GHz frequency for both split options 2 and 7, utilization was under 0.5 implying that less than half of core is utilised effectively.

For both split options, effective core utilization decreased with an increase in CPU frequency. The difference in effective utilization between split options 2 and 7 at a particular frequency is due to different components running at the DU side i.e., for split option 2, it is running RLC, MAC, and PHY at the DU side and the RRC and PDCP at the CU side. Whereas for split option 7, low PHY is running at the DU

://www.overleaf.com/project/5ee935e1f9fc620001b1fd7ax

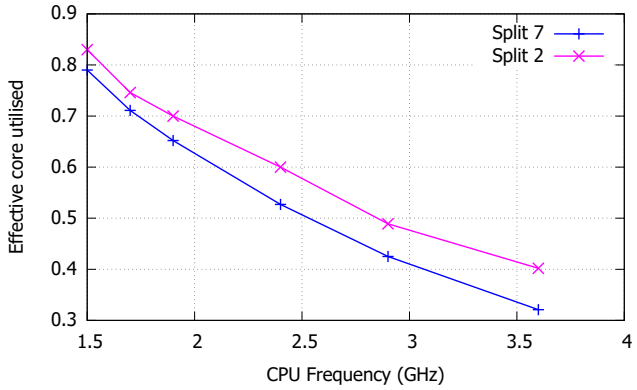

Fig. 11. CPU utilization for splits 2 and 7 with downlink traffic

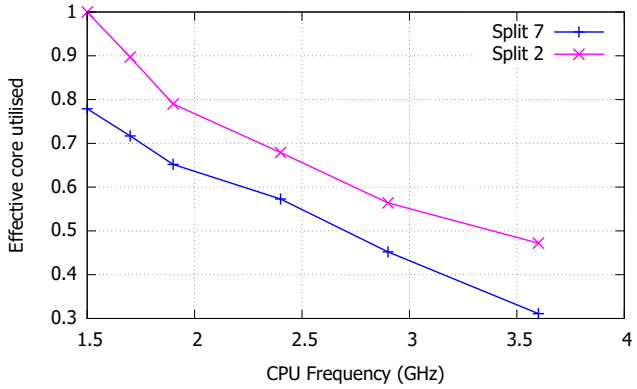

Fig. 12. CPU utilization for splits 2 and 7 with uplink traffic

side and RLC, MAC, and High PHY are running at the CU side.

C. Front-end and Back-end Bound

Front-end bound metric represents a pipeline slot fraction where the processor's front-end under supplies its back-end. Front-end denotes the first part of the processor core responsible for fetching operations that are executed later on by the back-end part. Back-end bound metric represents a pipeline slot fraction where no micro-operations are delivered due to lack of the required resources for accepting new micro-operations in the back-end. Experiments are conducted to study the effect of change in frequency on front-end and back-end bound percentages. These percentage are not affected by change in CPU frequency for both the split options.

D. Ideal frequency for running CU and DU

In split 2 configuration using 25 PRBs, the ideal frequency configuration for CU and DU is at 1.7 GHz and 1.9 GHz, respectively for full-buffer traffic scenario. Similarly, in split 7 configuration using 50 PRBs, the ideal frequency for CU and DU is 1.9 GHz and 1.7 GHz, respectively. In the case of 50 PRBs for both the splits 2 and 7, the ideal frequency of both CU and DU is 2.4 GHz resulting in maximum energy savings without compromising anything in terms of system throughput.

V. Conclusions

With the help of an experimental C-RAN testbed setup using OAI platform, energy values were profiled for different functional splits at multiple channel bandwidths by varying the clock frequency for both uplink and downlink traffic. Irrespective of the split option used, it was observed that for 25 PRBs, the CPU clock frequency could be scaled down to 1.7 GHz for some cases. Similarly, for the system running at 50 PRBs, the CPU clock frequency could be scaled down to 2.4 GHz. It was further observed that the system could run on two active cores with no loss in the throughput at the UE. Another notable observation was that there was no degradation in throughput and user experience even after reducing the number of cores.

With this information, operators can incorporate these factors into their C-RAN deployments. The CPU frequency and number of active CPU cores could be scaled for different available bandwidths and traffic loads. Therefore, with the given advantage of reducing the overall system energy, the remaining inactive cores can be utilized for processing other kinds of loads in the underlying cloud platform like MEC applications. By utilizing less number of cores for any split, multiple different splits with different bandwidths on existing hardware can be run. This significant finding can help reduce some of the CAPEX/OPEX costs.

Acknowledgment

This work has been supported by the project "Converged Cloud Radio Access Networks" funded by Intel India.

References

[1] J. Wu, Z. Zhang, Y. Hong, and Y. Wen, "Cloud radio access network (C-RAN): a primer," IEEE Network, vol. 29, no. 1, pp. 35–41, 2015
[2] China Mobile Research Institute, "C-RAN: The Road Towards Green RAN," White Paper, Sept. 2013.
[3] A. I. Salama and M. M. Elmesalawy, "Experimental OAI-based Testbed for Evaluating the Impact of Different Functional Splits on C-RAN Performance," 2019 Novel Intelligent and Leading Emerging Sciences Conference (NILES), Giza, Egypt, 2019, pp. 170-173, doi: 10.1109/NILES.2019.8909310.
[4] T. X. Tran, A. Younis and D. Pompili, "Understanding the Computational Requirements of Virtualized Baseband Units Using a Programmable Cloud Radio Access Network Testbed," 2017 IEEE International Conference on Autonomic Computing (ICAC), Columbus, OH, 2017, pp. 221-226, doi: 10.1109/ICAC.2017.42.
[5] P. Lin and S. Huang, "Performance Profiling of Cloud Radio Access Networks using OpenAirInterface," 2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Honolulu, HI, USA, 2018, pp. 454-458, doi: 10.23919/APSIPA.2018.8659532.
[6] 3GPP, "R3-161813-Transport requirement for CU-DU functional splits options."
[7] EURECOM, "Open air interface." Available: http://www.openairinterface.org/,
[8] "Perf", Available: https://perf.wiki.kernel.org/
[9] Intel,"Vtune-Profiler" Available: https://software.intel.com/content/www/us/en/develop/tools/vtune-profiler.html
[10] "Processor Counter Monitor (PCM)" Available: https://github.com/opcm/pcm