Experimental Evaluation of Power Consumption in Virtualized Base Stations

Ayala-Romero, Jose A.; Khalid, Ihtisham ; Garcia-Saavedra, Andres; Costa-Perez, Xavier; Iosifidis, George

DOI
10.1109/ICC42927.2021.9500323

Publication date
2021

Document Version
Final published version

Published in
ICC 2021 - IEEE International Conference on Communications, Proceedings

Experimental Evaluation of Power Consumption in Virtualized Base Stations

Jose A. Ayala-Romero*, Ihtisham Khalid*, Andres Garcia-Saavedra†, Xavier Costa-Perez†‡, George Iosifidis§
*School of Computer Science and Statistics, Trinity College Dublin, Ireland
†NEC Laboratories Europe, Germany
‡ i2CAT Foundation and ICREA, Spain
§Delft University of Technology, Netherlands

ICC 2021 - IEEE International Conference on Communications | 978-1-7281-7122-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICC42927.2021.9500323

Abstract—**Network virtualization is intended to be a key element of new generation networks. However, it is no clear how the implantation of this new paradigm will affect the power consumption of the network. To shed light on this relatively unexplored topic, we evaluate and analyze the power consumption of virtualized Base Station (vBS) experimentally. In particular, we measure the power consumption associated with uplink transmissions as a function of different variables such as traffic load, channel quality, modulation selection, and bandwidth. We find interesting tradeoffs between power savings and performance and propose two linear mixed-effect models to approximate the experimental data. These models allow us to understand the power behavior of the vBS and select power-efficient configurations. We release our experimental dataset hoping to foster further efforts in this research area.**

I. Introduction

Next generation mobile networks are intended to cope with an increasing traffic load [1] coming from new demanding applications [2]. A promising method for accommodating these needs is network densification: deploy more base stations (BSs) in order to shrink the cell size, offer high-throughput links to users, and reuse efficiently the wireless spectrum [3]. This will change dramatically cellular networks, which will comprise orders of magnitude more BSs, with different size and technology. Indeed, nowadays there is a fast shift from hardware-heavy BSs to smaller softwarized BSs [4]. Prominent examples are the open-source platforms `OpenAirInterface` [5] and `srsLTE` [6], but also proprietary initiatives [7]. These BSs implement their functions using software, which offers unprecedented management flexibility and supports their *virtualization* in shared hardware [8].

However, this network densification and softwarization comes at a cost for the environment and operators' budget. In particular, energy consumption accounts for 15-30% percent of network OPEX in developing markets [9], and 70% of that is attributed to BSs [10]. It becomes thus clear that the successful deployment of next generation networks depends highly on being able to answer the following questions: i) How much power do such virtualized base stations consume? ii) What parameters affect their power consumption? iii) How can we reduce their power costs? Some of these questions have been asked in the past, yet previous studies do not offer valid answers for this context due to two main reasons.

First, the BSs softwarization poses a completely new architecture. While legacy BSs operate with dedicated hardware, the baseband units (BBU) of virtualized BS (vBSs) are implemented in software that runs in general purpose processors. This raises questions about their power consumption, as previous studies have considered totally different equipment. Moreover, while in legacy BSs the transmission power is a predominant power cost factor, the network densification implies lower transmission power, hence lessening its overall importance. In this context, the power that is consumed at the BBU becomes a significant contributor in the overall power budget of vBSs, which motivates our experimental work.

Second, the computational load of the BBU changes with several parameters that affect differently the power consumption. In particular, the authors in [11] show that computational load at the BBU presents non-linear relations with the channel quality, the modulation and coding scheme (MCS) and the traffic load; yet it is not clear how these can affect the power consumption. In contrast, this behavior does not appear in legacy BSs. For instance, measurements in [12] show that the power consumption only varies up to 3% when the traffic intensity goes from no load to peak level.

Motivated by the above, we deploy a bespoke testbed based on `srsLTE` [6] and use extensive experiments to shed light on the relatively unexplored topic of *power consumption of virtualized base stations*. We focus mainly on the CPU power consumption at BBU, which is the main different component from legacy BSs, and we also measure the total vBS power (computing platform and radio). We study the uplink because its characterization is more intricate than the downlink, i.e., the power-load relation is non-linear and involves 2.5 times more computations. Also, the uplink is significant in today's networks where uploading user-generated content is common[1].

We first study how the SNR and airtime (or duty cycle) affect the CPU power consumption. We observe linear relations between the consumed power and these parameters; and propose a statistical model based on our measurements that are collected in real-time and for various system configurations. Previous works analyzed the impact of traffic on power, but there are no studies regarding the SNR or airtime. We find,

[1]For instance, live streaming communications (e.g., YouTube Live), content uploading in video platforms, and data exchange in social networking platforms.

interestingly, that different combinations of SNR and airtime lead to the same throughput *yet, different* power consumption; this creates room for power-aware vBS configurations, and we provide several examples.

Going a step further, we explore how the MCS selection affects the CPU power; and we discover a tradeoff between throughput and power consumption. We identify non-linear relation between the MCS and the parameters studied above (SNR and airtime). The reason behind this is that when SNR decreases, more computational resources are needed to decode the signal – as we measure in a set of experiments here. Based on these results, we propose a holistic power consumption model as a function of SNR, airtime, and MCS. This model can be used to design a power-efficient (at the expense of throughput) or throughput-efficient (at the expense of power) MSC radio scheduler; and we demonstrate this by modifying the `srsLTE` scheduler and measuring its performance.

We measure and compare the CPU power with the total power consumption. We find that indeed, the processing accounts for up to 50% of the total power. We also assess different computing platforms, ranging from small units (such as in small vBSs) to larger servers (as in cloud-RAN type of architectures), and also for different bandwidths. We see that our models are qualitatively robust, but naturally require adjustments in their coefficients when the systems are substantially different. And, finally, we create a fully documented experimental dataset available to the community [13].

II. Related Work

Legacy BS Power Consumption Models. Previous works proposed power consumption models for legacy BSs [14]–[20]. These models capture mainly the effect of power amplifier, RF output, baseband processing, and losses; while fewer include factors for microwave link, number of sectors, and transceivers. The work [14] proposed the seminal *EARTH* model which maps the RF output power to the total supplied power of a BS; and [15] integrated the transmission bandwidth in that model. The works [16], [17] proposed power models for macro and micro BSs, specifically, and [18] studied how the packet length affects the CPU power consumption. Power measurements of legacy BSs (GSM and UMTS) of daily traffic patterns are shown in [19]. A more detailed, non-linear, power consumption model w.r.t. the different BS components is presented in [20]. These models are related to downlink transmissions and hardware BSs. Although for those devices the transmission power is a predominant power contributor, for the new generation of small form-factor vBSs other parameters are equally important. For example, 40% of power consumption in femtocells is due to baseband processing [14]. Hence, it is imperative to revisit the models and enrich them with factors that affect the processing (such as the SNR).

Consumption Model for vBS Power. A computational-aware power consumption model for vBSs is presented in [21]. It suggests that the EARTH model cannot be generalized for vBSs because it does not cater for the vBSs computing intricacies. The authors propose a theoretical model of CPU power consumption as a function of the active CPU cores, clock speed, and load. It assumes a linear relation of data rate with the computational load, and hence with the consumed power. Although we have assessed this linearity for a general case in our work, we have also found configurations where this relation is not linear. In fact, we show later that the same data rate can be achieved with different power consumption, depending on the configuration of other variables. This non-linear relation is also shown in [11].

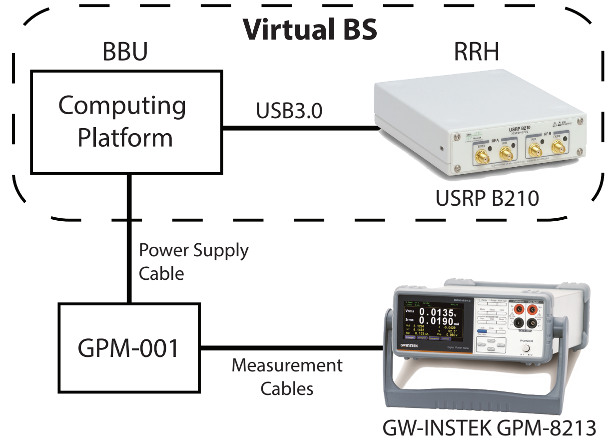


Fig. 1: **Connection scheme of the vBS and the power meter.** The vBS comprises an RRH (Ettus Research USRP B210) and a computing platform with a general purpose CPU. The RRH is connected to the platform using a USB3.0 and is fed through it. The power supply cable of the BBU is connected to GPM-001 measuring adapter, which feeds the BBU and enables the power meter GPM-8213 to measure its power consumption. The UE comprises another USRP B210 and a computing platform (not shown).

Baseband Processing Time & Complexity. Other works have studied the effect of MCS, bandwidth, and SNR on the computational load at the BBU [11], [22], [23]. An OpenAirInterface (OAI) based framework to pool compute resources of multiple BSs is presented in [22]. Measurement results show processing time to be linear with MCS and bandwidth, and uplink processing load to be 2.5 times higher than the downlink, due to turbo decoding. In [23] an OAI simulator is used to model uplink and downlink baseband processing time for different MCS, bandwidth, and virtualization platforms. A computing complexity analysis in cloud based radio access networks for centralized baseband processing of uplink signals is presented in [11]. These works characterize properly the BBU computation needs; however, the relation between the computing load and power consumption is not clear and needs to be measured and modeled.

III. Power Measurement Testbed

Testbed architecture. Our testbed is depicted in Fig. 1 and comprises the virtual base station (vBS), the user equipment (UE), and a digital power meter. The vBS consists of the Remote Radio Head (RRH) for which we use the Ettus Research

USRP B210[2], and the baseband unit (BBU) implemented in a computing platform. In order to evaluate the impact of hardware, we use two small factor PCs (Intel NUCs) and also two servers (see Table I). The RRH and BBU are connected through a USB3.0 cable, which means that the RRH is fully fed by the BBU (no external power supply). For the UE, we also use an Ettus Research USRP B210 and a general purpose computing platform. The vBS and UE are directly connected using SMA cables with 20dB RF attenuators. We use the power meter GW-Instek GPM-8213 along with the GW-Instek Measuring adapter GPM-001. From the available open-source 4G LTE stack, we consider `srsLTE` because as compared to `OpenAirInterface` it provides considerable gains in terms of CPU execution time, memory requirements, and stability for higher bandwidth [24]. We use the version 19.12 of `srsLTE` and Ubuntu 18.04 in all computing platforms.

Power measurements. We measure the vBS power consumption via software *and* hardware. For the former, we use the Intel's Running Average Power Limit (RAPL) functionality integrated into the Linux kernel to measure the CPU power. RAPL provides several counters indicating power consumption information using software power models. These models estimate the power consumption by using hardware performance counters and I/O models. Some works have assessed the accuracy of RAPL measurements showing that, in most cases, they match the actual power values [25]–[27]. We obtain the RAPL measurements using the Linux program `turbostat`. We also measure the vBS power via hardware with the GPM-8213 meter connected as explained above. Note that, when we measure the power via software, we are only considering the *CPU power*. On the other hand, the measurement via hardware includes the *power of the entire platform* (CPU, motherboard, RAM memory, etc.) plus the radio part, since the RRH is fed through the USB3.0 cable connected to BBU.

Experimental procedure. For the generation of our dataset [13], we set the configuration of the vBS and UE for a time period of 1 minute. We take samples of power consumption continuously via software and hardware during this period, and compute their average and variance. For each sample in our dataset, we configure the bandwidth, Transmission Mode (TM), uplink traffic load, transmission gain at the UE (which directly impacts the SNR), Modulation and Coding Scheme (MCS) and Airtime. The traffic load is generated using `mgen`[3].

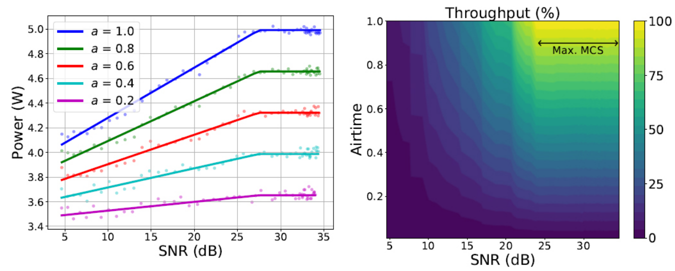
We use a customized version of `srsLTE` with which we change the MCS and airtime through a TCP socket on the fly. The vanilla `srsLTE` version only allows to change the MCS on the configuration file (needs restarting the vBS) and the airtime configuration is not supported. Our customized version also includes another TCP socket to get the performance of the vBS on the fly using JSON format. A Python script selects the radio configuration and gathers the measurements in a centralized way. The measurements of the power meter are also gathered from Python using the SCPI (Standard Commands for Programmable Instruments) interface through USB2.0. Finally, unless otherwise stated, the experimental data of the figures shown in the paper corresponds to the device NUC1 configured with a bandwidth of 10MHz.

[2]https://kb.ettus.com/B200/B210/B200mini/B205mini.

[3]https://www.nrl.navy.mil/itd/ncs/products/mgen.

Alias	Commercial name	CPU
NUC1	BOXNUC8I7BEH	i7-8559U @ 2.70GHz
NUC2	NUC7i7DNHE	i7-8650U @ 1.90GHz
Server1	Dell XPS 8900 Series	i7-6700 @ 3.40GHz
Server2	Dell Alienware Aurora R5	i7-9700 @ 3.00GHz

TABLE I: Computing platforms considered in the experimental evaluation. All the CPUs have been manufactured by Intel.



(a) (b)

Fig. 2: (a) Power consumption in the processor as a function of the SNR for different values of airtime. The dotted values represents the experimental values and the full lines are given by our model defined in Eq. (1). (b) Percentage of the maximum throughput as a function of the airtime and SNR.

IV. Impact of SNR and Airtime on Power

In this section, we propose the first model which gives us the power consumption of the Uplink in a vBS as a function of the SNR and airtime. We can easily compute the airtime from the traffic demand and MCS. Namely, the airtime is the percentage of subframes needed to support the traffic given the instant data rate[4]. For this model, we use the default radio scheduler of `srsLTE`, which selects the MCS for each given measured channel quality.

Fig. 2a shows the measurements of the CPU power consumption (scattered dots). We observe that the power consumption grows linearly with SNR. This is because higher SNR allows the use of higher MCS, which in turn induces more decoding computational load [11]. Recall that the mapping of SNR to MCS is performed by the `srsLTE` scheduler. We also observe that after a certain SNR value (approx. 28 dBs) the power remains constant. The reason is that no higher MCSs are selected after this point, and therefore the computational load is not further increased. Fig. 2a also shows the power consumption for different airtime values. Reducing the airtime not only implies the reduction of the constant values of the curves (the consumed power for the highest SNR) but also of the curves' slope.

Based on the behavior we have observed in the experiments, we propose a linear mixed-effect model [28] that can capture

[4]For example, if the instant data rate is 20 Mbps and the traffic demand is 15 Mbps, the airtime is $a = 0.75$.

the power consumption as a function of SNR and airtime. First, we denote $c \in \mathbb{R}^+$ as the SNR in dBs and $a \in [0,1]$ as the airtime, where $a = 1$ indicates that all subframes are used, and $a = 0$ indicates zero throughput. The CPU power consumption P is given by:

$$P(a,c) = \begin{cases} P_0(a) - r(a)\cdot(\gamma_0 - c) & \text{if } c < \gamma_0 \\ P_0(a) & \text{otherwise} \end{cases} \quad (1)$$

where $P_0(a)$ and $r(a)$ are the maximum power for a fixed airtime value and slope of the power consumption curve, respectively, which are defined as follows:

$$P_0(a) = \gamma_1 + \gamma_2 \cdot a, \qquad r(a) = \gamma_3 + \gamma_4 \cdot a. \quad (2)$$

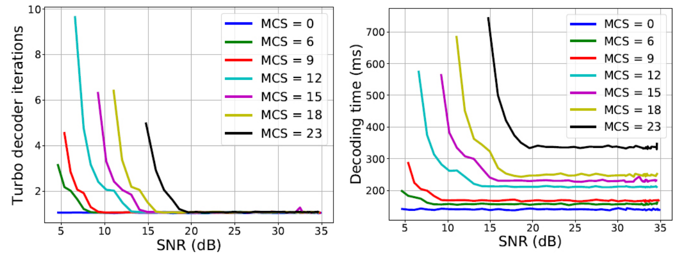
Fig. 2a shows the model (full lines). The values of $\boldsymbol{\gamma} = (\gamma_0, \ldots, \gamma_4)$ vary depending on several factors such as the bandwidth and computing platform, and are obtained using the Least Squares Method (LSM) in our dataset. The values of $\boldsymbol{\gamma}$ for different computing platforms and different bandwidths can be computed using the dataset and the source code we made publicly available [13].

Fig. 2b shows the joint impact of airtime and SNR on throughput. We observe that, while the throughput decreases linearly with the airtime (y-axis), on the x-axis it depends on the MCS assigned to each SNR value. Although the same throughput can be achieved by using different combinations of SNR and airtime as Fig. 2b shows, higher SNR and lower airtime will always reduce the power consumption of the vBS in this setting. However, achieving a high SNR can be costly in terms of energy for the UE in some cases (increase of its uplink transmission power). Therefore, the UE may decide to reduce its transmission power to save energy. This decision will imply an increment of the power consumed by the vBS, as our experiments show. Our detailed models can be used to fine-tune all these parameters to achieve the selected performance in terms of power consumption at the vBS, throughput, and energy costs at user devices. As our experiments reveal, there are several non-trivial tradeoffs in this setting.

V. Impact of MCS Radio Scheduler on Power

In the previous section, we studied the impact of SNR and traffic load on power consumption. However, these findings were conditioned on the scheduler used by `srsLTE`, which implements a certain rule for selecting the MCS based on the measured channel quality. If one can *redesign the scheduler*, which indeed is possible in open-source platforms, there are new opportunities for optimizing power consumption. To the best of our knowledge, this is the first work that experimentally evaluates the effect of MCS selection on power.

The default scheduler of `srsLTE` decides the MCS of each user based on its channel quality and depends on: i) the computation of the Channel Quality Indicator (CQI), and ii) the mapping between CQI and the maximum coderate. While the latter is standardized ([29], Table 7.2.3-1), the CQI computation is not defined in the specification, nor it is clear which factor should be involved on that. `srsLTE`, implements



(a) (b)

Fig. 3: Turbo decoder iterations and decoding time as a function of the SNR for different MCS values.

the mapping SNR to CQI from [30], and this aspect is open to new implementations. For that reason, we propose a new power consumption model including the MCS as a configuring variable, making this model suitable for any specific scheduler.

However, not all MCSs are feasible for any SNR value. The higher the MCS the less noise is tolerable during decoding. In addition, the computational load of decoding increases when the SNR is reduced. This is because the turbodecoder in the BBU needs more iterations for lower SNR values, which implies higher decoding time. We have done a separate series of experiments to measure this effect, presented in Fig. 3a-3b, which are in line with previous works, e.g., [11].

As we saw in the previous section, the power consumption depends on the airtime and the MCS selected by the radio scheduler. Now, we observe in Fig.3b that for each MCS $m \in \mathbb{Z}$ there is an SNR $c_{\text{th}}(m)$ below which the computational load starts increasing. This has a direct impact on power consumption. Furthermore, we model the slope of the power consumption increase with $r(a,m)$ since we observe in our data that it depends on the MCS and airtime. Based on these facts, we propose the following mixed-effect model:

$$P(a,c,m) = \begin{cases} P_0(a,m) + r(a,m)\cdot c & \text{if } c_{\min}(m) < c < c_{\text{th}}(m) \\ P_0(a,m) & \text{if } c > c_{\text{th}}(m) \end{cases} \quad (3)$$

where a is the airtime and c the SNR, and we define:

$$P_0(a,m) = \beta_0 + \beta_1 \cdot a + \beta_2 \cdot m + \beta_3 \cdot a^2 + \beta_4 \cdot m^2 + \beta_5 \cdot a \cdot m \quad (4)$$

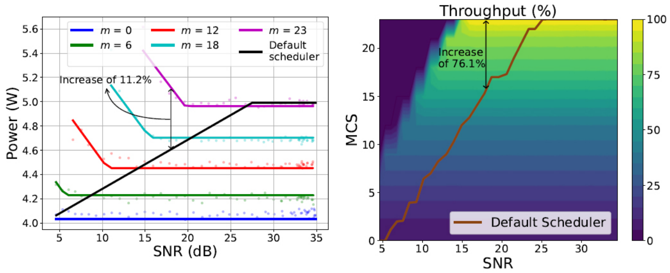
$$r(a,m) = \beta_6 + \beta_7 \cdot a + \beta_8 \cdot m \quad (5)$$

$$c_{\min}(m) = \beta_9 + \beta_{10} \cdot m \quad (6)$$

$$c_{\text{th}}(m) = \beta_{11} + \beta_{12} \cdot m. \quad (7)$$

The power $P(a,c,m)$ is not defined for $c < c_{\min}(m)$ since for these points the combination of m and c is not feasible. Similarly to Sec. IV, the value of $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_{12})$ is fitted using LSM and depends on the computing platform and radio bandwidth (discussed in Sec. VI).

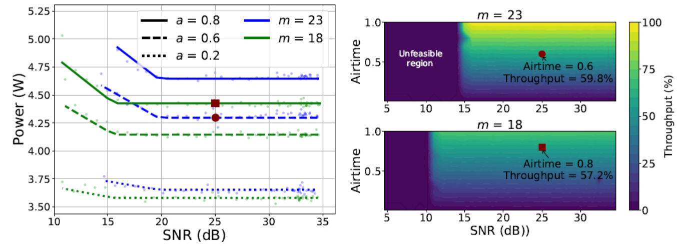
Fig. 4a shows the CPU power consumption as a function of SNR and MCS, for airtime $a = 1$ (full buffer). The scatter points correspond to our measurements and the full lines are given by Eq. 3. We have also included the power consumption of the default `srsLTE` scheduler (modeled in the previous



(a)

(b)

Fig. 4: (a) Power consumption in the processor as a function of the SNR for different MCSs and airtime equals 1. The dotted values represent the experimental values and the full lines are given by our model. In black, we show the power consumption of given by the model in the previous section, in which the MCS configuration is given. (b) Percentage of the maximum throughput as a function of the MCS and SNR for airtime equals 1. In brown, we show the MCS assignment of `srsLTE` default scheduler.



(a)

(b)

Fig. 5: (a) Power consumption in the processor as a function of the SNR for different MCS and airtime values. (b) Percentage of the maximum throughput as a function of the airtime and SNR for the selected MCSs.

section) for comparison. Fig. 4b depicts the throughput as a function of SNR and MCS for full buffer. The zero values in the upper left corner indicate the unfeasible combinations of SNR and MCS, i.e., points at which the SNR is not high enough to decode a specific MCS (decoding error). The brown line indicates the MCS values selected by the default `srsLTE` scheduler for a given SNR. We observe in Fig. 4a-4b a tradeoff between power and throughput. For instance, in Fig. 4a and for SNR c= 25 dB, the `srsLTE` scheduler consumes 4.85W, while if we select MCS $m = 23$ the consumption is 5.12W (11.2% power increase). Fig. 4a shows the impact of these decisions on throughput that can be increased by 76.1% when we select the maximum MCS value instead the default one.

Fig. 5a shows the dependency of power consumption on the airtime for two MCS values. We observe that the minimum feasible value of SNR for each MCS ($c_{\min}(m)$ in our model) and the inflection point from which the power starts increasing ($c_{\text{th}}(m)$ in our model) do not depend on airtime. This is captured in Eq. (6)-(7). Moreover, as we mentioned in the previous section, the airtime has a direct impact on power.

Fig. 5b depicts the effect of airtime on the throughput for two selected MCS values. The markers in this figure indicate two configurations with similar throughput. These two configurations are also marked in Fig. 5a in which we can observe that, although achieving the same throughput, there is a small difference in power consumption. This is more prevalent when the difference between the MCSs is larger. This indicates that, when the channel quality is good, the use of higher MCSs is more efficient in terms of power. However, if the SNR is reduced to 15 dBs (following the same example), this changes, i.e., the configuration with the lower MCS consumes less power with the same throughput. This shows again the non-linear relation between the SNR and power consumed at the BBU, which can be properly exploited based on the network's priorities for performance or costs.

VI. Computing Platform Evaluation

In Sec. IV we proposed a model for the power consumption of the uplink as a function of SNR and airtime, which was extended in Sec. V to include the radio configuration (MCS). Our experiments show that both the CPU power and the total power follow the patterns shown in Fig. 2a-4a. The reason is that we are evaluating the uplink which is very computationally intensive. Only control signals are sent through the downlink in our experiments and therefore the transmission power has a reduced impact on the overall power. This means that the proposed power consumption model can approximate CPU and total power by selecting the proper values of γ (first model) and β (second model) for each case.

Moreover, the power consumption exhibits qualitatively the same behavior for all computing platforms and bandwidths. However, since we are using general purpose processors the power will change depending on the energy requirements of each CPU, its architecture, etc. As an example, for NUC1 and bandwidth 10 MHz we have $\boldsymbol{\gamma}$ =(27.49, 3.355, 0.522, -0.0011, 0.0414), $\boldsymbol{\beta}$ =(3.324, 0.731, -0.008, -0.024, 0.00037, 0.039, 0.753, -0.822, 0.005, 0.097, -0.00034) for the power consumed by the CPU and $\boldsymbol{\gamma}$ =(26.83, 10.5, 0.85, -0.001, 0.072), $\boldsymbol{\beta}$ =(10.51,1.296,-0.011, -0.184,0.00053,0.0613, -0.5063, 0.93,0.027,0.136, -0.0017) for the total power of the vBS. The values of these parameters for all the computing platforms in Table I and for several bandwidths can be easily computed using the code provided along with our dataset [13].

Fig. 6 shows the total power consumption and the power consumption of the CPU for 4 computing platforms and 3 bandwidth values when the vBS is operating with full buffer and high SNR. We consider two architectures: i) two small factor PCs (NUCs) with expected reduced power consumption and diverse processors with different energy requirements, and ii) two general purpose servers.

First, we observe that power consumption increases up to 460.86% when we change from a small factor PC to a general purpose server. Second, the selection of the bandwidth of the vBS is the paramount importance and should be selected accordingly to the network requirements to avoid not only the waste of energy but also the inefficient use of the radio resources. We see an increase of 20.27% on average when we

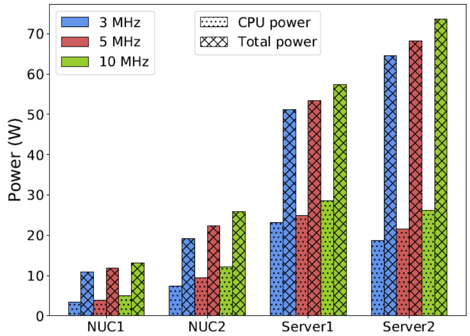


Fig. 6: Comparison of power consumption for 3 bandwidths and several computing platforms, using high SNR and full buffer.

change from 3 MHz to 10 MHz. Third, even using a platform with the same form-factor we find important differences in the CPU and total power consumption, e.g., NUC2 consumes 86.93% more than NUC1 on average. Finally, we observe that the portion of the power consumed by the CPU with respect to the total power is very remarkable and goes between 28.96% and 49.69%.

VII. Conclusions

We studied experimentally the power consumption in virtualized base stations. We built a testbed to measure the power consumption in real time, using `srsLTE`, a general purpose computing platform for the BBU, and a USRP for the RRH. We focus on the less explored uplink power consumption problem because its characterization is more challenging than the downlink since it presents non-linear relation between the load and resource usage. We first measure the power as a function of SNR and airtime. We observe (and measure) that the power increases with the SNR, since higher MCSs are used with more favorable channels, which in turn induce more computational load. We identify a tradeoff between the power consumption of the UE and vBS. The UE can save energy by reducing the transmission power. However, this will worsen the SNR increasing the power consumption of the vBS.

Second, we evaluate the power consumption as a function of the MCS. In that case, we find non-linear relations between the SNR and the power consumption. The reason is that when the channel quality deteriorates, the received signal is more noise and the turbo decoder in the BBU needs more iterations to decode the signal. This implies an increase in the decoding time and therefore of the power consumption. We observe that, for a certain objective throughput, we can find different configurations with different power consumption. That is, when the channel quality is favorable configurations with higher MCS and lower airtime can reduce the consumed power. In contrast, for lower SNR values it is more efficient in terms of energy to use lower MCSs and higher airtime.

Finally, we propose two linear mixed-effect models to approximate our experimental data with which characterize several computing platforms with different energy requirements. We release our experimental dataset hoping to foster further efforts in this research area.

Acknowledgments

This work was supported by the European Commission through Grant No. 856709 (5Growth) and Grant No. 101017109 (DAEMON); and by SFI through Grant No. SFI 17/CDA/4760.

References

[1] V. Cisco, "Cisco visual networking index: Forecast and trends, 2017–2022," *White Paper*, vol. 1, 2018.
[2] Alliance, NGMN, "5G white paper," *Next generation mobile networks, white paper*, vol. 1, 2015.
[3] J. Andrews *et al.*, "What will 5g be?" *IEEE JSAC*, vol. 32, no. 6, 2014.
[4] I. Afolabi *et al.*, "Network slicing and softwarization: A survey on principles, enabling technologies, and solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, 2018.
[5] N. Nikaein *et al.*, "OpenAirInterface: A flexible platform for 5G research," *ACM SIGCOMM CCR*, vol. 44, no. 5, pp. 33–38, 2014.
[6] I. Gomez-Miguelez *et al.*, "srsLTE: an open-source platform for LTE evolution and experimentation," in *in Proc. of ACM WinTech*, 2016.
[7] A. Inc., "ECOMP (Enhanced Control, Orchestration, Management & Policy) Architecture," *White paper*, Mar. 2016.
[8] P. Rost *et al.*, "Mobile network architecture evolution toward 5g," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 84–91, 2016.
[9] "5G network energy efficiency," Nokia Corporation. White Paper, 2016.
[10] T. Edler and S. Lundberg, "Energy efficiency enhancements in radio access networks," *Ericsson review*, vol. 81, no. 1, pp. 42–51, 2004.
[11] P. Rost *et al.*, "The complexity–rate tradeoff of centralized radio access networks," *IEEE Trans. on Wirel. Comm.*, vol. 14, 2015.
[12] A. Corliano and M. Hufschmid, *Energieverbrauch der mobilen Kommunikation: Schlussbericht*. Bundesamt für Energie, Feb. 2008.
[13] J. A. Ayala-Romero *et al.*, "Dataset - Experimental evaluation of power consumption in virtualized base stations," 2021. [Online]. Available: https://github.com/jaayala/power_ul_dataset
[14] G. Auer *et al.*, "How much energy is needed to run a wireless network?" *IEEE Wireless Communications*, vol. 18, no. 5, pp. 40–49, 2011.
[15] H. Holtkamp *et al.*, "A parameterized base station power model," *IEEE Communications Letters*, vol. 17, no. 11, pp. 2033–2035, 2013.
[16] O. Arnold *et al.*, "Power consumption modeling of different base station types in heterogeneous cellular networks," in *IEEE Future Netw. & Mob. Summit*, 2010.
[17] M. Deruyck *et al.*, "Power consumption model for macrocell and microcell base stations," *Trans. on Emerg. Telecomm. Techn.*, 2014.
[18] B. H. Jung *et al.*, "Modeling of power consumption for macro-, micro-, and rrh-based base station architectures," in *Proc. of IEEE VTC Spring*. IEEE, 2014.
[19] J. Lorincz *et al.*, "Measurements and modelling of base station power consumption under real traffic loads," *Sensors*, vol. 12, 2012.
[20] B. Debaillie *et al.*, "A flexible and future-proof power model for cellular base stations," in *Proc. of IEE VTC Spring*, 2015.
[21] T. Zhao *et al.*, "Energy-delay tradeoffs of virtual base stations with a computational-resource-aware energy consumption model," in *Proc. of IEEE ICCS*, 2014.
[22] S. Bhaumik *et al.*, "CloudIQ: A framework for processing base stations in a data center," in *Proc. of ACM Mobicom*, 2012.
[23] N. Nikaein, "Processing radio access network functions in the cloud: Critical issues and modeling," in *Proceedings of the 6th International Workshop on Mobile Cloud Computing and Services*, 2015, pp. 36–43.
[24] Z. Geng *et al.*, "Performance analysis and comparison of GPP-based SDR systems," in *Proc. of IEEE MAPE*, 2017.
[25] E. Rotem *et al.*, "Power-management architecture of the intel microarchitecture code-named sandy bridge," *IEEE micro*, vol. 32, 2012.
[26] D. Hackenberg *et al.*, "Power measurement techniques on standard compute nodes: quantitative comparison," in *IEEE ISPASS*, 2013.
[27] A. Mazouz, B. Pradelle, and W. Jalby, "Statistical validation methodology of cpu power probes," in *Proc. of ECPP*. Springer, 2014.
[28] A. Gelman and J. Hill, *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press, 2006.
[29] 3GPP TS 36.213, "Evolved Universal Terrestrial Radio Access (E-UTRA) Physical layer procedures (Release 8)," *V9.0.1*, (Dec. 2009).
[30] M. Kawser *et al.*, "Downlink SNR to CQI mapping for different multipleantenna techniques in LTE," *Intern. Jrl. of Inform. and Electr. Eng.*, vol. 2, no. 5, 2012.