Maximising the Energy Efficiency of Virtualised C-RAN Via Optimising the Number of Virtual Machines

Raad S. Alhumaima, Riyadh Khalf Ahmed, and H.S. Al-Raweshidy[1]

1234914@my.brunel.ac.uk

***Abstract*—In cloud radio access networks (C-RAN), more accurate prediction of the number of virtual machines (VMs) one server can support would improve network capacity and energy efficiency (EE). In this paper, the problem of allocating an optimal number of VMs to the cloud server is introduced. Monte Carlo based evolutionary algorithm (PSO, QPSO or GA) are used to find the suboptimal number of VMs that optimises the energy efficiency (EE) of C-RAN. To enable such evaluation, a power model is proposed to evaluate the power consumption (PC) of each unit within a virtualised server. This evaluation occurs under the circumstances of increased number of hosted VMs, and processed resource blocks (RBs) at each VM. Moreover, power allocation methods are proposed to transmit the power from base band unit (BBU) pool to the remote radio heads (RRHs), and from RRHs to the users (UEs). This allocation is based on the combination of one or more of RRH distance, RRH channel gain, UE distance, UE channel gain, and UE path loss. The EE problem was constrained to crucial quality of service (QoS) indicators, including minimum UE data rate, number of allocated RBs, and latency imposed due to virtualisation.**

Index Terms*—Virtualisation, optimisation, BBU pool, Power Allocation, cloud radio access networks, energy efficiency, virtual machines

I. Introduction

Driven by the need to provide at least 10 times higher spectral and energy efficiency (EE) in 5G networks, mobile operators deployed a large number of small cells in heterogeneous networks. Whilst this has increased network capacity, it has also led to the consumption of more power. In order to reduce this consumption, cloud-radio access network (C-RAN) architecture is proposed [1], [2]. In C-RAN, the base band units (BBUs) servers are responsible for processing the upper layers and most of the physical layer functions, including radio frequency (RF), base band digital signal processing, and wireless media access control (MAC) functions. These BBUs are cloudified in a centralised location, called a BBU pool. Consequently, the remote radio heads (RRHs) are left exceedingly simple at the cell site, with only optical to electrical conversions, amplifiers, and antennas [3]. C-RAN contrasts the traditional long term evolution (LTE) system, where the BBU functions are processed in the evolved NodeB (eNodeB) at the cell site itself. Bringing these BBUs together in C-RAN has resulted in a paradigm that is capable of effectively implementing advanced cooperation algorithms, utilising the spectrum by using dynamic bandwidth adaptation approaches, exploiting the load variation to reduce the required cooling and total power consumption/consumed (PC), and adapting new 5G enabling technologies [4]. Additionally, C-RAN diminishes operational expenditures (OPEX) and capital expenditures (CAPEX) due to reduced maintenance cost and fewer site visits [5]. Despite the C-RAN paradigm unleashing the network potentials greatly regarding EE and cost of operation, intensifying the number of deployed RRHs and active BBUs leads to considerable increases in the amount of PC [3]. Hence, the goal of offering a highly efficient 5G architecture that is able to decrease this PC substantially is a challenge. In order to meet it, the research community has embraced the use of network function virtualisation (NFV) techniques in the cloud. In fact, both C-RAN and NFV represent the key success technologies in the coming generations. C-RAN offers low cost system, higher level of efficient resources sharing, while virtualisation technology can provide major reductions in PC. Moreover, the service providers (SPs) have gained the ability to allocate network resources flexibly within the cloud. In addition, there has been automation in the virtualised server's operation and configuration, reduced maintenance cost, and support for multi-tenancy mode of service. On top of this, NFV has allowed deploying and managing new services to fulfil the UEs' demands on the fly. It has also led to the promotion of the concept of hardware-software isolation in the virtualised servers [6], [7], which allows for the execution of network functions using only software, called virtual machines (VMs). These VMs can run on general off-the-shelf servers, rather than proprietary built or dedicated appliances [8].

VMs are expected to reduce the operational cost in the core networks. Today, it is possible to operate fewer virtualised servers in the pool to run the whole network while fulfilling the UE quality of service (QoS) requirements [9]. Each VM is software that runs the BBU functions and shares the resources of the host server with other VMs in a time restricted manner. Running multiple VMs on a single hardware requires a HyperVisor (HV), which is software that runs on the server's higher layer, thus allowing the host to be shared by multiple VMs [10]. That is, each VM can utilise the server's random access memory (RAM), central processing unit (CPU), network interface cards (NICs) and hard drive (HDD) by itself without obstructing other VMs. First, the HV collects the information of each VM regarding the number of UEs and their QoS indicators, subsequently scheduling the available

resource blocks (RBs) amongst these VMs. Afterwards, each VM can schedule its share of RBs amongst its UEs according to different QoS factors such as, minimum data rate, received power, interference, etc. However, the presence of HV within the host server increases RAM accesses, CPU functional complexity and HDD usage, with eventual consequent extra overhead occurring within the virtualised server. Furthermore, the existence of VMs increases the host server's latency. Hence, a detailed comparison of the virtualised and bare servers is required to identify the advantages and disadvantages of using NFV in C-RAN.

A. NFV Trade-offs

1) NFV is able to curtail the increase in PC in C-RAN due to integrating new technologies and services such as, software defined network (SDN) and load balancing appliances [3]. This reduction can be achieved by sharing the available server's resources/units while cascading multiple VMs in one operating server.
2) It was mentioned in [11], that a single virtualised server can host tens of VMs. However, this might be possible when the server is running offline applications, the delay of which is relaxed. Because a virtualised server with 1 VM may take about 5 times more execution time delay to process a packet compared to its traditional counterparts [12], such delay in online services can not be avoided. This delay originates from each VM only being able to own a scarce amount of the host server's resources as it has to share these with other VMs. Clearly, if the server hosts more VMs, the resources share allocated to each VM is further reduced [13]. In this case, the VM is obligated to queue its load and wait for a window to be opened again by the HV after a time. Hence, optimising the number of installed VMs in one host server is prerequisite, so that the VMs can always meet their real time requirements, and ensures the host server is not overloaded.
3) It was measured in [14], that the execution time of a traditional base station's functions is convexly or linearly proportional to the number of processed RBs. This means when the VM operates as a BBU, increasing the number of its allocated RBs will surely produce extra delay. Whilst the total number of VMs generate an enlarge amount of delay, the allocated RBs of each VM have to be optimised.
4) Finally, the virtualised server itself gains a PC as its resources are fully utilised. The reasons of such consumption is due to higher computation levels, generated I/O instructions, and compound accessing for the device resources by the aggregated VMs' applications. Consequently, a busy virtualised server may consume about 40% more power than traditional counterparts [12]. This increament requires further investigations regarding modelling the PC of virtualised servers.

The above contradictions galvanise the estimation of what is the optimum number of VMs for sustaining the network's QoS. This simple question leads to the generation of further inquiry, such as, what is the amount of overhead these VMs draw upon? How does increasing the number of processed RBs at each VM can affect the PC and latency of the host server? What is the highest latency that can be avoided by the network? What is the optimal number of allocated RBs to each VM? What is the minimum QoS requirement for each UE served by a particular VM? These questions originate differentiated network variables including RBs, PC, delay and data rate. Hence, we have assembled these parameters to be correlated in an EE problem.

B. Main Contributions

1) The optimal number of VMs that maximises the EE of C-RAN has been estimated. The problem is solved using particle swarm optimisation (PSO), quantum PSO (QPSO) and genetic algorithm (GA) approaches.
2) We intended to measure the amount of traffic (number of UEs and RRHs, power allocation, channel gains, etc.) found in the area of interest, and examines a possible change in the traffic volume prior to optimisation process. For this purpose, a Monte Carlo method was adapted inside a PSO, QPSO and GA algorithms to assume large number of possible network traffic. This method is different to what is found in the available literature, where the network is constantly adapted to a new solution each time the traffic volume or network behaviour is changed.
3) In contrast to the uniform distributions of the UEs and RRHs, which are based on hexagonal, circular or triangular shape. In this work, Poison Point Process (PPP) distribution has been used to reflect on the real-time deployment and practical-wise resources assignments. These included randomly generating the RRHs and UEs positions, power, resource blocks scheduling, etc, for each Monte-Carlo iteration.
4) Modelling the way active VMs and utilised RBs increasingly affect the PC of the host servers. This modelling provides a realistic evaluation to the PC modelling at the server's unit level, i.e. CPU, RAM, NIC and HDD. A well-known LTE parameter (i.e. RB) has been used as a main factor in this modelling. Eventually, this parameter can affect both the sum rate and PC during the optimisation process.
5) The baseband signals transmitted from the BBU pool towards the fiber/wireless connected RRHs are distributed over the proposed power allocation methods: PAM1, PAM2 and direct power. These allocations are based on either UE distances to the RRH, both these distances and channel gain, or equal power distribution to the RRHs. Note that direct power method means the RRHs are allocated equal power from the pool. However, these allocations are different than in the available research works, in which the sum data rate is directly influenced by only the resources assignments in the fornthaul (from RRHs to the UEs). However, in C-RAN, both fronthaul and backhaul (resources assignment from BBU pool to RRHs) should be considered. Accordingly, the proposed methods in (1) and (2)) correlate the distances (from BBU pool to the RRHs) with the RRH's received power, such that the pool can have an influence up on the latter. Subsequently, these RRHs impact upon UEs' received power and their data rates.

C. Related Work

In this section, both sides of the EE problem are discussed, namely, the workload management in virtualised data centres,

and PC modelling. A comprehensive survey in [15] reviewed most of the available PC models to date, including virtualised and non virtualised servers, data centres as well as single server. Generally, the available power models can be classified into hardware, software intrusive and machine learning. This classification is based on the approach that has been used in the measurement. Intrusion based models are required to install intrusive hardware tools and events counters, which makes the PC measuring expensive and complex. Moreover, these measuring hardware counters add additional PC overhead to the actually measured PC. This runs against the generic purpose of the research, which is aimed at reducing the PC as much as possible. It was observed that the main parameter to be measured in these models is the utilisation ratio/level that is scored by the CPU, RAM or storage during operation. Measuring such a factor requires there to be a server, from the operating system of which this value can be monitored, and this clearly increases cost along with complexity. The authors in [16] have proposed to track the VM's energy usage at each hardware unit via using HV-observable hardware power states. The software based models are similar to the first method, but this type use a monitoring software, which is installed as an application on the server so that the VMs' power usage can be known. This process is also complex, the installed software can be a reason for more PC within the server. Furthermore, the tracking process cannot guarantee accurate measurement for the events that occur, because of the time response mismatch between these high frequency events and the time window opened to track them, see [17]. The third method is based on machine learning or heuristics and is error acceptable as it is based on random distribution of the solution candidates. Also, it requires repeating the process of optimisation several times to guarantee the results, for example see [18]. Furthermore, this method is time consuming and costly on power. However, our proposed power model is much simpler and costless when compared to the available models, it is only based on the number of hosted VMs, allocated bandwidth/RBs to each VM, and components data sheets.

On the other hand, there have been several works with the aim to optimising the EE in the data centres. In [19] and [20], a dynamic, on-demand VM migration based algorithms were proposed for distributed data centres. These works were devoted to reducing the carbon footprint based on specified service level agreement (SLA). The authors in [21] put forward a live migration technique amongst the cloud servers to adapt dynamically the load fluctuating. In [22], an energy efficient algorithm that reduces the operational costs in virtualised data centres was suggested. The algorithm consolidates VMs based on current CPU utilization using live migration technique. However, in these works, there was no evaluation for the migration power cost, not to mention the increased delay within the virtualised server. The power cost can reach up to 32W in the source, and 10W in the destination server for each migrated VM [23]. If these numbers are multiplied by the number of migrated VMs, such a price would militate against the deployment of these methods. In [24], a technique was proposed to reduce the electricity bill through allocating the coming traffic amongst distributed, internet based, and non virtualised data centres. Authors in [25] were concerned with optimising the energy cost in the data centres. Their proposed algorithm adapts the traffic demand over time to reduce the power. This work places emphasis on service and infrastructure providers, and their revenue to satisfy a certain SLA. The authors of [26] suggested using both, the traditional power grid and renewable energy to reduce the PC in data centres. Through a time varying and traffic adaptation based algorithm, they proposed allocating some of the network tasks to the renewable energy sources. However, the cost of renewable energy was not evaluated, such as maintenance, deployment and gain over traditional source of energy. In [27] and [28], the authors put forward an approach to reduce the carbon footprint by redirecting the traffic to cleaner geographical locations. In [29], the authors proposed an algorithm that splits the coming traffic to different data centres instead of one destination, with the main objective being to balance the coming workload prior to processing. A highly related work to our problem can be found in [30], where the number of VMs a server can support is experimentally assessed. Unfortunately, there has been no UE resources allocation, no power model, and no mathematical representation to be able to generalise such a case to broader amount of server types.

II. System Model

The downlink multi-RRH, multi-UE C-RAN system included total number of RRHs (M). These RRHs are PPP distributed with intensity (λ_1). On the other hand, the total number of UEs (U) are distributed with intensity (λ_2). Each RRH (m) is assumed to have a sub-number of UEs (U_m). The nearest distance-based UEs then attached to the RRH m and distributed with coordinates (x_u, y_u), with small scale fading h that is assumed to be Rayleigh fading. The noise power is assumed to be additive with a value of (σ^2). Furthermore, each UE (u) holds an Euclidean distance ($d_{m,u}$) to the serving RRH m, where $d_{m,u} = \sqrt{(x_m - x_u)^2 + (y_m - y_u)^2}$. The RRHs are positioned at coordinates (x_m, y_m), each RRH is located with Euclidean distance ($d_{m,o}$) to the BBU pool, where $d_{m,o} = \sqrt{(x_m - x)^2 + (y_m - y)^2}$. The pool in turn is positioned at $(x = 0, y = 0)$ at the centre of the geographical area.

A. Optical Power Allocation Models

Two methods are proposed to distribute the power from BBU pool to the optical fibre, with star topology connected RRHs. The first method or power model (PAM1) is a distance-based proportional allocation, where RRHs received power relies on the distance $d_{m,o}$ to the BBU pool. That is, the closer RRH m is to the BBU pool, the less power (Pr_m) it will receive as compared to other RRHs, as follows:

$$Pr_m = \frac{(Ppool - OFL)\ d_{m,o}}{\sum_{m=1}^{M} d_{m,o}} \tag{1}$$

Where OFL denotes the fibre losses and $Ppool$ denotes the total power transmitted by the BBU pool. In traditional or partially centralised networks, the need for such allocation (i.e. from BBU pool to RRHs) is ignored, because the BBU unit

already resides within the eNodeB, and transmits the signals to the UEs through RF unit. In addition, the connection from eNodeB to the core network is only logical via the transport links. However, with fully centralised C-RAN, the BBU unit is shifted to the BBU pool. Hence, the modulated signals are no longer generated at the eNodeB, but rather, from the BBU pool. To describe this relocation, power allocation methods from the BBU pool to the RRHs are planned.

The second model (PAM2) is proposed based on both the RRH-BBU pool distances and the channel gain received by the U_m-th UEs. If an increase within the total channel gain of UEs (U_m) is taking place. Alongside, the RRH is more distant to the BBU pool when compared to other RRHs, this situation allows an increase within the power received (Pr_m) by the tagged RRH m in comparison to other RRHs. Additionally, the RRH's received power can be further disciplined through the power control variable (δ) which sets the effectiveness of this proportional power distribution. This method can be introduced as follows:

$$Pr_m = \frac{(Ppool - OFL)\ (h_u^m\ d_{m,o})^\delta}{\sum_{m=1}^{M}\sum_{u=1}^{U_m}(h_u^m\ d_{m,o})^\delta} \tag{2}$$

Where $u \in \{1, ..., U_m\}$ is the UE index, $(0 \leq \delta \geq 1)$ is the power allocation effectiveness control factor, and $h_u^m = |h|^2$ is the signal attenuation of u-th UE within m-th RRH.

Consequently, the u-th UE can be allocated an amount of power based on three methods. The first allocation is based on the distance, where the u-th UE is allocated this according to its distance ($d_{m,u}$) when compared to other UEs distances within the m-th RRH, as follows:

$$P_{m,u}^n = \frac{(Pr_m)\ (d_{m,u})^\delta}{\sum_{m=1}^{M}\sum_{u=1}^{U_m}(d_{m,u})^\delta} \tag{3}$$

The second allocation is based on both, the distance ($d_{m,u}$) and the received channel gain ($h_u^{m,n}$) compared to other UEs within the m-th RRH, as follows:

$$P_{m,u}^n = \frac{(Pr_m)\ (h_u^{m,n}\ d_{m,u})^\delta}{\sum_{m=1}^{M}\sum_{u=1}^{U_m}(h_u^{m,n}\ d_{m,u})^\delta} \tag{4}$$

Where $P_{m,u}^n$ and $h_u^{m,n}$ denote UE's received power and channel gain from m-th RRH served by n-th VM. Moreover, the third allocation is based on both the path loss (r_u^m) and the small scale fading ($h_u^{m,n}$) [31], as follows:

$$P_{m,u}^n = Pr_m\ h_u^{m,n}\ r_u^m \tag{5}$$

Where $r_u^m = (d_{m,u})^{-\alpha}$ indicates the path loss from the RRH m to UE u, and α is path loss exponent. Subsequently, the sum data rate can be given as:

$$Csum = \sum_{m=1}^{M}\sum_{n=1}^{N}\sum_{u=1}^{U_m}\sum_{rb=1}^{RB_n} B_o \log_2(1 + P_{m,u,rb}^n \sigma_{m,u,rb}^n) \tag{6}$$

Where RB_n represents the total number of RBs allocated to VM n with bandwidth B_o, $P_{m,u,rb}^n$ is the allowed transmitted power on RB (rb), and $\sigma_{m,u,rb}^n$ represents the SINR of rb served by VM n assigned to UE u of RRH m, where

$$\sigma_{m,u,rb}^n = \frac{P_{m,u,rb}^n\ h_{u,rb}^{m,n}\ r_u^m}{B_o N_o + I_{du}} \tag{7}$$

and

$$I_{du} = \sum_{inf \in \phi/m}^{Inf} h_u^{inf}\ r_u^{inf} \tag{8}$$

I_{du} is the aggregate interference from all other interferers RRHs (Inf) excluding the serving RRH m. Moreover, $r_u^{inf} = (R_u^{inf})^{-\alpha}$ stands for the path loss from the interferer RRH (inf) to the UE u, R_u^{inf} is the distance of interferer RRH inf to the UE u, and h_u^{inf} is the channel gain of interferer RRH inf to the UE u. It is worth mentioning that maximisation of the sum bit rate of all UEs does not guarantee this for each individually. Hence, the bit rate of each UE is constrained to a threshold value, as presented in (10).

In regards to the PC, there are four major participants involved within the constituency of a server, these are RAM, CPU, NIC and HDD. It was mentioned in [32] and [33] that the PC of the virtualised server is exponentially or non linearly proportional to the number of VMs. Hence, the PC of the virtualised server (P_{srvr}) can be expressed as $P_{srvr} = (P_{ram} + P_{cpu} + P_{nic} + P_{hdd}) \times e^{\varepsilon N}$, where P_{ram}, P_{cpu} P_{nic} and P_{hdd} denote the initial PCs of server's RAM, CPU, NIC and HDD, respectively. The term ($e^{\varepsilon N}$) is used to describe the dynamic PC of the server's units due to the existence of VMs, where ε is a positive constant. Since each VM is serving several UEs, it is assumed that the dynamic load or bandwidth share is linearly proportional to the number of processed RBs [14]. This means the more UEs served at each VM, the more RBs that are processed, which increases the dynamic or traffic based consumption as a greater share of the finite server resources is demanded. Consequently, the total number of processed RBs in a server ($RB_T = \sum_n^N RB_n$) is added to P_{srvr} to assemble the total consumption of a server ($P_{server} = P_{srvr} + e^{\vartheta_n * RB_T}$). RB_n denotes the total number of RBs processed by each VM, and ϑ_n is the increment factor due to processing RB_n by VM n in any of the server resources. These RBs are concerned with adding an important decision weight to both sides of the EE formulation.

Another performance factor is the time it takes the VM to process these RBs. The execution time of the workload in a traditional BBU server increases linearly with both the number of RBs and the modulation coding scheme ($MCS \in \{9, 16, 25\}$) that is used to transmit/receive these RBs [14]. In a virtualisation environment, a single VM requires π times more delay to process a packet compared to the traditional counterparts. This is due to increased accessing calls and interrupts among VM-HV and HV-server's unit, where π can reach up to 5 [12]. Modelling this concept requires introducing a factor called MCS index (mcs) to describe the linear relationship between the RBs and execution time in a bare BBU server (τ_{bare}), where $\tau_{bare} = \tau_{init} + (mcs * RB_n)$, τ_{init} denotes the initial BBU delay due to other BBU functions, rather than MCS. Furthermore, the HV delay (π) is added to the above description to produce the execution time of virtualised server (τ_v^n) when 1 VM is found in the server, i.e., $\tau_v^n = \tau_{bare} + \pi$.

Subsequently, the total execution time of all VMs (τ_{vms}) is produced by jointly adding τ_v^n of all available VMs, where $\tau_{vms} = \sum_n^N \tau_v^n$.

B. Gain of Virtualisation

With a 10 MHz bandwidth, there are 50 RBs available at each 0.5 ms, or 100 RBs per transmission time interval (TTI), also called (Subframe=1 ms). Whilst the minimum allocated resources to a single UE is 12 sub-carriers in one TTI, which is equivalent to 2 RBs in the time domain. Eventually, the BBU can serve up to 50 UEs each millisecond. If 100 UEs are connected, the scheduler takes at least 2 ms to serve them all. This logic is correct, but commercially it is difficult to design such scheduler to handle 50 UEs in 1 ms, because there are a minimum number of RBs assigned to each UE in a certain TTI to guarantee its minimum QoS. This means there can be more than 2 RBs assigned to the UE in each TTI. In a virtualised server, the total number of scheduled RBs in one TTI can reach up to $N \times BB$, where BB denotes the number of traditional BBU servers. This is because each VM performs as a separate traditional BBU device through a software abstract. Amongst the VMs, the HV is responsible for managing these available RBs, where each VM n is assigned a certain number of (RB_n), according to their load, the UE's channel condition, the UE's distance, etc. On the other hand, this increment in the number of RBs that are required to be processed in one TTI imposes another speculation regarding whether there is any available server capable of serving such a number (i.e, $N \times BB$)?. In answer to this, the current advances in hardware manufacturing show that a single BBU server is capable of processing up to 900 LTE UEs [34]. In traditional network operation, this number can barely be reached, as in actual server performance, not all UEs are active at the same time. However, in virtualisation environment, this offers some non-utilised server resources that can be exploited by the VMs. Hence, such a situation allows each VM to process its allocated RBs on time. Eventually, this facilitates a reducing in the total number of bare servers, which diminishes the consumed power and improves the EE, without compromising the network performance.

III. Total Power Consumption

Total PC of the virtualised server is also subjected to the effects of other losses such as, AC-DC, DC-DC and cooling loss in a straight forward manner. These losses is linearly scaled with other components' PC and approximated by using loss factors (σ_{DC}, σ_{AC}, σ_{cool}) to represent AC-DC, DC-DC and cooling, respectively [35]. Successively, the total PC of virtualised C-RAN (P_{vCRAN}) is modelled as the combination of virtualised cloud BBU server's PC ($\frac{P_{server}+P_{opt,P}}{(1-\sigma_{DC})(1-\sigma_{MS})(1-\sigma_{cool})}$), and RRH's PC ($P_{RRH}$) which is modelled as ($\frac{(Pt_m/\eta_{PA})+P_{RF}+P_{opt,R}}{(1-\sigma_{DC,R})(1-\sigma_{MS,R})}$). Moreover, the RRH transmitted power (Pt_m) is equivalent to its received power (Pr_m) if no power gain is added. $P_{opt,P}$ and $P_{opt,R}$ denote the PC of optical devices in the BBU pool and the RRH, respectively. $\sigma_{DC,R}$ and $\sigma_{MS,R}$ denote RRH's DC and RRH's MS loss factors, respectively. Moreover, there will be no cooling offered to the RRH. Finally, P_{RF} is RF unit's PC, $\frac{Pt_m}{\eta_{PA}}$ is the PC of power amplifier (PA), and η_{PA} is its efficiency.

IV. Problem Formulation

The Sum EE of vC-RAN system is defined as how much sum data rate can the UEs receive in one Watt. The formulation of such problem is described as follows:

$$\max \quad \text{EE (N)} = \frac{\text{Csum}}{\text{P}_{\text{vCRAN}}} \tag{9}$$

$$\text{S.t.} \quad Csum^n_{m,u,rb} \geq Csum_{thr}, \ \forall \ u, rb \tag{10}$$

$$\tau_u + \tau_m + \tau_{vms} \leq \tau_{thr} \tag{11}$$

$$\tau_{vms} \leq \tau^{thr}_{vms} \tag{12}$$

$$RB_n \leq 100, \ \forall \ n \tag{13}$$

$$\sum_{n}^{N} RB_n \leq RB_T \tag{14}$$

$$\sum_{u}^{U_m} \sum_{rb}^{RB_n} P^n_{m,u,rb} \leq Pr_m, \ P^n_{m,u,rb} \geq 0, \ \forall \ u, rb \tag{15}$$

$$\sum_{m}^{M} Pr_m \leq Ppool, \ \forall \ m \tag{16}$$

Where $Csum^n_{m,u,rb} = B_o \log_2(1 + P^n_{m,u,rb}\sigma^n_{m,u,rb})$, $Csum_{thr}$ is the minimum QoS requirement. The second constraint (11) represents the round trip latency restriction, where $\tau_u = 2 \times \arg\max(\frac{d_{m,u}}{sol})$ pertains the round trip signals latency of the most distant UE u served by RRH m. In addition, $\tau_m = 2 \times \arg\max(\frac{d_{m,o}}{v})$ denotes the maximum round trip latency of the signals travelling from RRH m to the BBU pool. Furthermore, $v = \frac{sol}{ind}$ holds the speed of light (sol) inside the optical fiber, and ind is the refractive index of the optical fiber, which is assumed to be identical for all RRHs-BBU pool links. Through the third constraint (12), the latency of the server due to one VM (τ_v^n) can be controlled and the latter as well as the fixed value of initial HV delay π can control the delay τ_{bare}. The latter, in turn, affects the number of processed RBs (RB_n) of each VM. By substituting the third constraint into the second, the total latency of the system will not exceed the latency threshold (τ_{thr}), where τ^{thr}_{vms} is the maximum latency threshold allowed to all VMs. Each VM n in constraint (13) can not exceed the maximum number of allocated RBs RB_n. Accordingly, constraint (14) deals with maximum number of RBs to be processed in the server by all VMs. Because recently the LTE servers' processing capability has become greater and more efficient, the maximum number of processed RBs (RB_T) is suggested as being 800. This number is shared amongst the VMs, each exploiting its allocated share to assign the required number of RBs to each UE, while satisfying other constraints. The sixth constraint (15) imposes the limitation regarding the power received by all UEs U_m on the total RBs RB_n. Finally, the constraint (16) indicates the total received power by all RRHs cannot exceed the total transmitted power of the BBU pool ($Ppool$).

To solve our problem, PSO, QPSO and a GA are used to search the solution space of a function to find the sub-optimal number of VMs (N) that maximises an objective/fitness function (EE) of C-RAN. The predominant issue is that the use of such algorithms holds a time constraint, which is the time needed to obtain the solution. However, this has been overcome in this work, as for a specific geographical area where the pool resides, a huge amount of potential traffic is considered. Specifically, for each Monte Carlo iteration, new UEs, RRHs, channel conditions and RB assignments are established by using PPP distribution. These iteration will cover, examine and expect all possible traffic situations in the area of interest on daily basis or for large periods of time. Hence, repeating the optimisation process each time the traffic is changed is no longer necessary. Another constraint is the sub-optimality of the given solution, which is also ignored, because, PSO, for example, yields a solution that is nearest to the optimal one. Since the required number of VMs is an integer, rounding the solution up and down will mitigates such behaviour, which holds true for QPSO and GA. Specifically, down scaling/flooring the solution variable (N) is preferred to prevent server over-load and thus, ensure its safe operation.

The reason of adapting PPP at each iteration, rather than treating uniformly, is to contemplate the practical deployment and real life scenarios during the distribution of RRHs and UEs [36]. The Poison distribution measures the probability that a certain number of events occur within a certain period of time. This stochastic process is one of the most important random processes in probability theory. It is widely used to model the random points in time and space, being an accurate way to model the spatial distribution of the geographical RRH location [37]. As it offers no constraint on the distances of the adjacent RRHs, it provides more realistic cell shape as well as SINR and EE measurements in comparison to the uniform distribution, as represented by hexagonal, circular, or triangle cell shapes. In PSO, the particles have random speeds through the solution space, each being assessed by an objective/fitness function with a best stored particle solution ($pbest$) and best stored overall solution ($gbest$). Based on the current particle's (i) position (N_i), its speed (v_i), its past best position ($pbest_i$) and best global position of whole particles ($gbest$), each individual particle is being updated interactively. PSO first initialises its particles or generations, with each particle representing a possible sub-optimal solution (the potential number of VMs), this possibility then undergoes the process described in Algorithm-1. In which, the possible particle solution (N_i) is subjected to the constraints.

Furthermore, at each particle evaluation, the Monte Carlo inner loop performs the following: (i) randomly generating the RRHs and UE assignments using PPP; (ii) repeating these steps R times of possibilities; and (iii) calculating the average sum EE of the UEs within the network, as shown in Algorithm 1. These steps will be repeated as many times as the number of particles (I). The reason of proposing the inner loop of Monte-Carlo with R iterations inside the main PSO algorithm with I particles, is to ensure that the solution is qualified for an enormous number of network formations, i.e ($R \times I$). For example, if $R = 1000$ and $I = 100$, this will produce ($1000 \times 100 = 100000$) possible RRHs and UE resources assignments, as at each particle i there are R iterations. Practically, this means that the resulting solution is valid for this number of network distributions. Indeed, if R is increased, this possibility approaches unity, and the number of covered network scenarios will be virtually closer to infinity. Eventually, this matter will strengthen the efficiency of the solution. However, this comes with increased execution time of the algorithm, which has been previously neglected. As such, this complexity is expected to increase when solving a paradigm with larger geographical area due to increasing the number of RRHs and UEs. PSO has a particular problem that arises from its structure of being a continuous algorithm. Consider a set of points such as {A,B,C and D}, in order for one particle to move from A to D, this requires to passing the points in between (B, and C). If these points are local minimums, there will be an inbuilt problem. As a solution, quantum PSO (QPSO) is proposed to follow a purely probabilistic scheme in which the next position is drawn from a probability distribution, thus having a discrete nature. QPSO was first proposed in [38] as a good complexity-performance trade-off method. It is based on both, the physical principle of quantum mechanics, and the social behaviour of swarms of various animals. In quantum theory, a qubit is the smallest unit of information, with its value relies in the range [0,1]. Each particle (i) holds quantum energy $q_i(t)$. The QPSO algorithm is similar to PSO, it stores the values of best position previously found for each particle $pbest$ and the global best position $gbest$. From these positions, the best global and individual quantum energy values are calculated in order to generate changes in the particle positions. The algorithm is implemented to our problem, as follows:

1) Generating the initial particles, each particle i with energy $q_i(t)$ is randomly generated at position $N_i(t)$.
2) Evaluating $N_i(t)$ through the cost function (9), if there is a better position for the particle i, the best individual and global positions will be updated.
3) Changing the energy $q_i(t)$ of each particle according to: $q_i(t+1) = c_1 q_i(t) + c_2 q_i^{best}(t) + c_3 q_g^{best}(t)$, where c_1, c_2, c_3 denote the weight of each component of energy, and $c_1 + c_2 + c_3 = 1$.
4) return to step 3 until reaching the total number of iterations [39].

Algorithm 1 : PSO main Algorithm

while (not terminating condition) **do**
Evaluate each particle
for particle i, $i = 1, 2, ..., I$ **do** (update the best positions)
if $EE(N_i) < f(pbest_i)$ **so** $pbest_i = N_i$
if $f(pbest_i) < f(gbest_i)$ **so**
$gbest_i = pbest_i$, **end if**, **end if**
for $r = 1 : R$
Evaluate sum $EE(N_{i,r})$
end for, **end for**
for particle i, $i = 1, 2, ..., I$ **do** (generate the next generation)
$v_i(t+1) = wv_i(t) + c1r1(pbest_i - (N_i)) + c_2r_2(gbest - (x_i, y_i))$,
$(N_i)(t+1) = (N_i)(t) + v_i(t+1)$
end for
end while

V. Results And Analysis

To correlate the findings of our problem with real-time scenarios, the resulting parameters were selected from [35], [40], [41], [42], [33], [43], as shown in Table I. The experimental data related to the PC of each component in the server demonstrate that initial PC of the CPU is 29.6W, RAM is 4W, NIC is 2W and HDD is 25W. Moreover, the rest of PC in the server is a result of the overhead, i.e. the AC-DC, DC-DC and cooling. Moreover, the parameters used in Table I led to about 40% PC increment within each virtualised server at maximum workload. This increment was real-time measured in [12], which represents the cost of over-utilising the server due to the existence of many VMs that share it's units. However, the proposed model is not constrained to only yielding this amount of percentage, but rather, is valid for any type of server through adjusting the model parameters. It is worth mentioning that different specifications of the server can affect the algorithm regarding the EE and resulting N, because each server might hold different manufacturing initials and efficiencies, which is required to be adjusted through PC initials and tuning factors such as, ε, ϑ_n, etc.

Fig. 1 shows the sum EE of C-RAN using only distance based power allocation from the RRHs to UEs (3). Additionally, the power from BBU pool towards the RRHs is distributed using three methods, these are PAM1 (1), PAM2 (2), and direct/equal power to all RRHs. In all cases, PSO is outperforming the GA; more information about the GA operation can be found in [44].

Moreover, Fig. 2 shows a comparison of sum EE using the same power allocation methods i.e., PAM1, PAM2 and direct power, but the RRHs-UE power allocation is based on distance and channel gain of (4). Based on the distance, the UEs are classified into center and edge. The RRH-center UEs are assumed to always have better channel conditions than RRH-edge UEs. Hence, the distance based technique allocates more power to the RRH-center UEs, which results in maximising the EE.

Furthermore, Fig. 3 shows the comparison of sum EE of C-RAN using PAM1, PAM2 and direct power, with the UEs having power allocated according to (5) [31]. Due to the existence of channel path loss within this method, it produces less EE performance when compared to the methods (4) and (3). Whilst the path loss degrades the power received by the UE, the SINR will be degraded, which results in less EE

Fig. 4 shows a comparison of sum EE of C-RAN by using the same power allocations of Fig. 2, but the system in not virtualised. Clearly, the non virtualised case has produced more PC, Hence, the network EE has been reduced. For further inquiry into this result, the traditional server consumption is produced. To achieve this, the effect of N and RBs has been removed from P_{vCRAN} formulation. If (P_{bbu}) symbolises the bare server's PC, where $P_{bbu} = [P_{ram} + (P_{cpu} \times K) + (P_{nic} \times L) + P_{hdd}]$, where K and L denote the total number of CPUs and NICs, respectively. Subsequently, the number of bare servers (BB) is multiplied by P_{bbu}. Afterwards, the amount $[(BB \times P_{bbu}) + P_{RRH}]$ has replaced the virtual case

TABLE I: PARAMETERS BREAKDOWN

Component	Unit	Value	Component	Unit	Value
λ_1	-	0.25	ϑ	-	0.001
λ_2	-	0.075	R	-	1000
eta_{PA}	-	0.36	ξ	-	0.007
ind	-	1.5	σ_{MS}	-	0.09
δ	-	0.9	P_{RF}	W	12.6
α	-	4	P_{nic}	W	2
RB_n	-	100	P_{cpu}	W	29.6
I	-	100	$P_{opt,R}$	W	1
σ_{cool}	-	0.1	τ_{vms}^{thr}	ms	6
$\sigma_{MS,R}$	-	0.09 τ_{init}	μ sec	80	
$\sigma_{DC,R}$	-	0.075	fr	MHz	2620
σ_{DC}	-	0.075	OFL	$\frac{dB}{Km}$	0.5
$mcs, 9$	-	6	$AWGN$	$\frac{dB}{Hz}$	-10
$mcs, 16$	-	9.5	BW	MHz	10
$mcs, 25$	-	17	$Ppool$	W	20
ν	-	0.005	$Csum_{thr}$	Kbps	10
c_2	-	1.2	P_{hdd}	W	10
c_1	-	0.2			

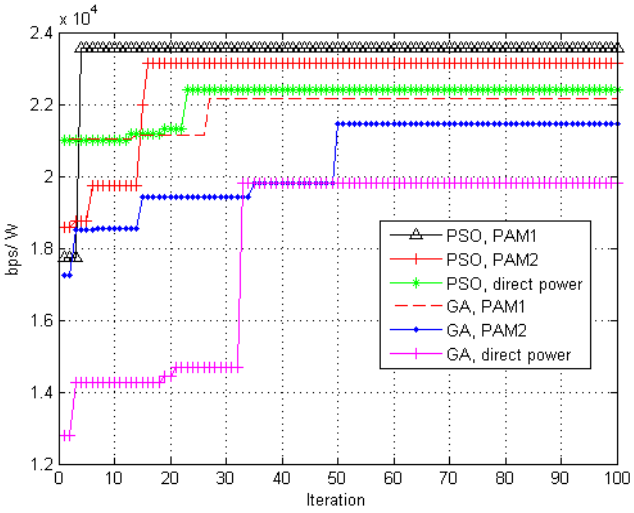

Fig. 1: EE comparison of virtualised C-RAN using PAM1, PAM2 and direct power allocation, the UEs are allocated power according to 3.

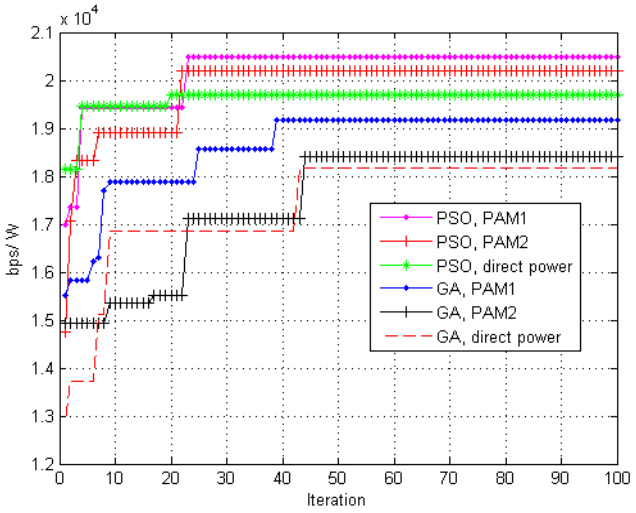

Fig. 2: EE evaluation of virtualised C-RAN using PAM1, PAM2 and direct power allocation, the users are allocated power using 4.

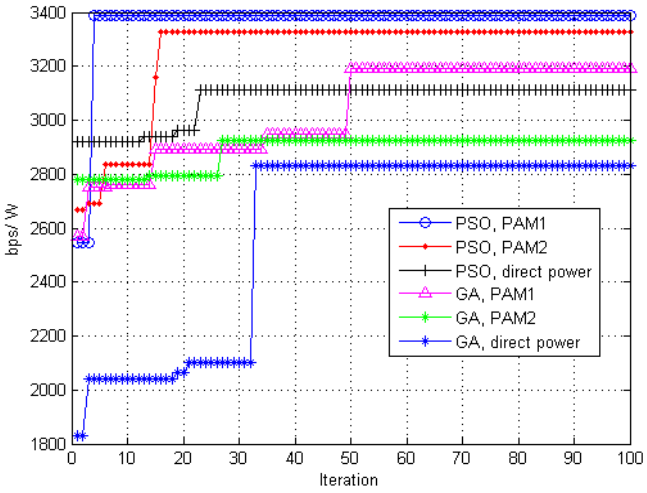

Fig. 4: EE evaluation of non-virtualised C-RAN using PAM1, PAM2 and direct power allocation.

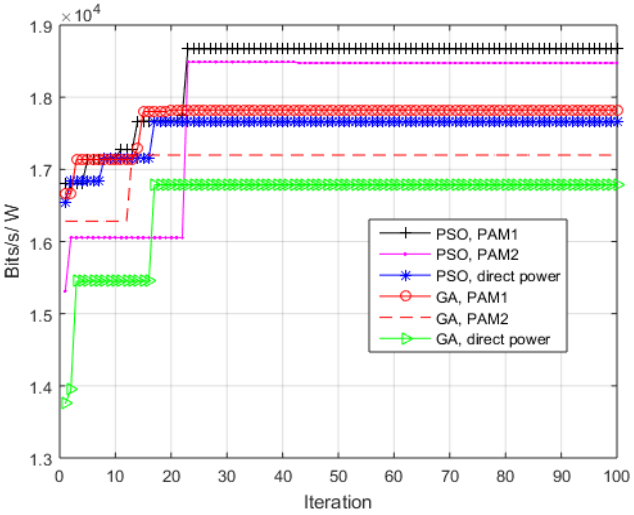

Fig. 3: EE evaluation of virtualised C-RAN using PAM1, PAM2 and direct power allocation, the UEs are allocated power according to (5).

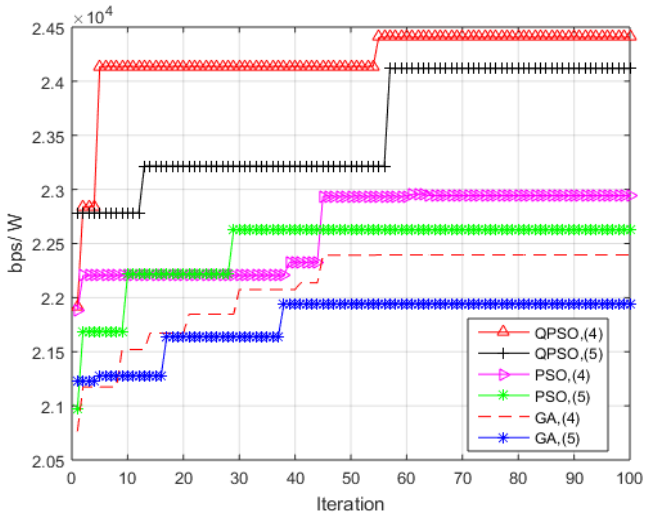

Fig. 5: EE evaluation of virtualised C-RAN using PSO, QPSO and GA, the UEs are allocated power according to (4) and (5), while RRHs are allocated power according to (2).

(i.e. P_{vCRAN}) in Algorithm 1, which upgrades the PC.

Moreover, Fig. 5 shows the sum EE performance comparison amongst QPSO, PSO and GA using UE power allocations of (4) and (5), while RRH-BBU pool allocation is based on PAM2, or (2). Due to channel path loss, the model of (4) constantly overcomes the EE output of model (5). Concurrently, QPSO algorithm performs better than the other (PSO and GA) algorithms, but with more convergence time due to its complex behaviour. However, all cases have resulted values of N in between 6 and 7. The selected tuning parameter of QPSO, PSO and GA are as follow: for PSO, the total number of particles I is equivalent to 100, inertia weight (w) is 0.9, the cognitive parameter (c_1) is 0.2, and social parameter (c_2) is 1.2. For GA, the number of generation is 100, the population size is 100 and the crossover probability is 0.8. Finally, QPSO parameters are selected according to [39]. All the results have been obtained after running the algorithms 20 times to overcome the randomness behaviour of heuristic algorithms. Then the run with highest record of each case has been selected. In all cases, PSO always converges faster than GA and QPSO, in about 23 generations, while GA constantly converges in more than 35 generations, and QPSO in more than 40 generations.

To understand further how the different parameters influence the EE outcome, we give a simple example starting with a single UE. The UE's received power is based on the RRH's power received from the pool. The RRH's received power affects the PA consumption, whilst the latter, in turn, influences server and total network consumption. Since the number of

VMs (N) relies within the formulation of total PC, this parameter is relatively affected. If there are no constraints, N tends to be zero, so as the PC is minimised, which maximises the EE. However, there are two effective constraints to prevent such failure. First, the total latency threshold, which binds the execution time to the restricted value in (11) and (12), and this accordingly, affects the resulting N. Second, the total number of RBs is involved in both the PC and sum rate calculations. When running the algorithm, the RBs aim to increase the sum rate of (9), whilst the same time decreasing the PC, because more processed RBs means more PC, as described in Section II.

VI. Conclusion and Potential Developments

The EE maximisation problem in virtualised C-RAN has been presented in the context of estimating the number of VMs that one server can support, without affecting the operating efficiency. To enable such an evaluation, a power model of virtualised server has been proposed to simulate the real time measurement. This model reflects the consequences of increasing the number of VMs found within the server, and processed RBs by each VM. In addition, the time constraint due to virtualisation technology is modelled as well as the execution time of processing the RBs in bare servers. This formulation is integrated with the total C-RAN's latency to participate in the optimisation process. While considering all the possible assignment in an area of interest using PPP oriented Monte Carlo method inside the main PSO, QPSO and GA algorithms, the network EE is evaluated. By adapting Monte Carlo, the necessity to repeat the optimisation process is avoided. At the same time, the long/short traffic variation problem has been overcome.

Multiple comparisons can be established when changing the way UEs receive their power. For example when using PSO, GA or pattern search algorithms instead of the power allocations of (3) and (4). However, the latter are proposed to relieve the run time. Finally, the provided mathematical representation in this work can be easily used to optimise the placement of the visualised BBU pool. By exploiting the distances amongst RRHs-BBU pool ($d_{m,o}$), the coordinates x and y can be considered as extra optimisation variables, instead of using $x = 0$ and $y = 0$ as reference values in this work. This can optimise the virtualised BBU pool position to guarantee a mitigated average delay among RRHs-BBU pool, reduced system PC and an enhanced EE.

References

[1] M. M. Mowla, I. Ahmad, D. Habibi, and Q. V. Phung, "A green communication model for 5g systems," *IEEE Transactions on Green Communications and Networking*, vol. 1, pp. 264–280, Sept 2017.

[2] C. Liu, L. Zhang, M. Zhu, J. Wang, L. Cheng, and G. K. Chang, "A novel multi-service small-cell cloud radio access network for mobile backhaul and computing based on radio-over-fiber technologies," *Journal of Lightwave Technology*, vol. 31, pp. 2869–2875, Sept 2013.

[3] R. S. Alhumaima, M. Khan, and H. S. Al-Raweshidy, "Component and parameterised power model for cloud radio access network," *IET Communications*, vol. 10, no. 7, pp. 745–752, 2016.

[4] J. Yao and N. Ansari, "Qos-aware joint bbu-rrh mapping and user association in cloud-rans," *IEEE Transactions on Green Communications and Networking*, pp. 1–1, 2018.

[5] M. Khan, R. S. Alhumaima, and H. S. Al-Raweshidy, "Reducing energy consumption by dynamic resource allocation in c-ran," in *2015 European Conference on Networks and Communications (EuCNC)*, pp. 169–174, June 2015.

[6] N. M. M. K. Chowdhury and R. Boutaba, "Network virtualization: state of the art and research challenges," *IEEE Communications Magazine*, vol. 47, pp. 20–26, July 2009.

[7] Z. Zhu, P. Gupta, Q. Wang, S. Kalyanaraman, Y. Lin, H. Franke, and S. Sarangi, "Virtual base station pool: Towards a wireless network cloud for radio access networks," in *Proceedings of the 8th ACM International Conference on Computing Frontiers*, CF '11, (New York, NY, USA), pp. 34:1–34:10, ACM, 2011.

[8] J. G. Herrera and J. F. Botero, "Resource allocation in nfv: A comprehensive survey," *IEEE Transactions on Network and Service Management*, vol. 13, pp. 518–532, Sept 2016.

[9] R. Riggio, A. Bradai, D. Harutyunyan, T. Rasheed, and T. Ahmed, "Scheduling wireless virtual networks functions," *IEEE Transactions on Network and Service Management*, vol. 13, pp. 240–252, June 2016.

[10] Z. Xiao, W. Song, and Q. Chen, "Dynamic resource allocation using virtual machines for cloud computing environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, pp. 1107–1117, June 2013.

[11] B. Pfaff, J. Pettit, K. Amidon, M. Casado, T. Koponen, and S. Shenker, "Extending networking into the virtualization layer.," in *Hotnets*, 2009.

[12] R. Shea, H. Wang, and J. Liu, "Power consumption of virtual machines with network transactions: Measurement and improvements," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, pp. 1051–1059, April 2014.

[13] R. Lent, "Evaluating the performance and power consumption of systems with virtual machines," in *Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on*, pp. 778–783, Nov 2011.

[14] S. Bhaumik, S. P. Chandrabose, M. K. Jataprolu, G. Kumar, A. Muralidhar, P. Polakos, V. Srinivasan, and T. Woo, "Cloudiq: a framework for processing base stations in a data center," in *Proceedings of the 18th annual international conference on Mobile computing and networking*, pp. 125–136, ACM, 2012.

[15] M. Dayarathna, Y. Wen, and R. Fan, "Data center energy consumption modeling: A survey," *IEEE Communications Surveys Tutorials*, vol. 18, pp. 732–794, Firstquarter 2016.

[16] A. Kansal, F. Zhao, J. Liu, N. Kothari, and A. A. Bhattacharya, "Virtual machine power metering and provisioning," in *Proceedings of the 1st ACM Symposium on Cloud Computing*, SoCC '10, (New York, NY, USA), pp. 39–50, ACM, 2010.

[17] X. Wu, C. Lively, V. Taylor, H.-C. Chang, C.-Y. Su, K. Cameron, S. Moore, D. Terpstra, and V. Weaver, "Mummi: multiple metrics modeling infrastructure," in *Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2013 14th ACIS International Conference on*, pp. 289–295, IEEE, 2013.

[18] J. L. Berral, Í. Goiri, R. Nou, F. Julià, J. Guitart, R. Gavaldà, and J. Torres, "Towards energy-aware scheduling in data centers using machine learning," in *Proceedings of the 1st International Conference on energy-Efficient Computing and Networking*, pp. 215–224, ACM, 2010.

[19] A. Amokrane, R. Langar, M. F. Zhani, R. Boutaba, and G. Pujolle, "Greenslater: On satisfying green slas in distributed clouds," *IEEE Transactions on Network and Service Management*, vol. 12, no. 3, pp. 363–376, 2015.

[20] M. F. Zhani, Q. Zhang, G. Simona, and R. Boutaba, "Vdc planner: Dynamic migration-aware virtual data center embedding for clouds," in *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, pp. 18–25, IEEE, 2013.

[21] M. F. Zhani, Q. Zhang, G. Simona, and R. Boutaba, "Vdc planner: Dynamic migration-aware virtual data center embedding for clouds," in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*, pp. 18–25, May 2013.

[22] A. Beloglazov and R. Buyya, "Energy efficient resource management in virtualized cloud data centers," in *Proceedings of the 2010 10th IEEE/ACM international conference on cluster, cloud and grid computing*, pp. 826–831, IEEE Computer Society, 2010.

[23] Q. Huang, F. Gao, R. Wang, and Z. Qi, "Power consumption of virtual machine live migration in clouds," in *Communications and Mobile Computing (CMC), 2011 Third International Conference on*, pp. 122–125, IEEE, 2011.

[24] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, "Cutting the electric bill for internet-scale systems," in *ACM SIGCOMM computer communication review*, vol. 39, pp. 123–134, ACM, 2009.

[25] Q. Zhang, Q. Zhu, M. F. Zhani, R. Boutaba, and J. L. Hellerstein, "Dynamic service placement in geographically distributed clouds," *IEEE*

Journal on Selected Areas in Communications, vol. 31, no. 12, pp. 762–772, 2013.

[26] D. Hatzopoulos, I. Koutsopoulos, G. Koutitas, and W. Van Heddeghem, "Dynamic virtual machine allocation in cloud server facility systems with renewable energy sources," in *Communications (ICC), 2013 IEEE International Conference on*, pp. 4217–4221, IEEE, 2013.

[27] P. X. Gao, A. R. Curtis, B. Wong, and S. Keshav, "It's not easy being green," in *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, pp. 211–222, ACM, 2012.

[28] Y. Guo, Y. Gong, Y. Fang, P. P. Khargonekar, and X. Geng, "Energy and network aware workload management for sustainable data centers with thermal storage," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 8, pp. 2030–2042, 2014.

[29] Y. Xin, I. Baldine, A. Mandal, C. Heermann, J. Chase, and A. Yumerefendi, "Embedding virtual topologies in networked clouds," in *Proceedings of the 6th international conference on future internet technologies*, pp. 26–29, ACM, 2011.

[30] A. Y. S. Tan, R. K. L. Ko, and V. Mendiratta, "Virtual numbers for virtual machines?," in *2014 IEEE 7th International Conference on Cloud Computing*, pp. 972–974, June 2014.

[31] H. H. Yang, J. Lee, and T. Q. S. Quek, "Heterogeneous cellular network with energy harvesting-based d2d communication," *IEEE Transactions on Wireless Communications*, vol. 15, pp. 1406–1419, Feb 2016.

[32] Y. Liao, L. Song, Y. Li, and Y. A. Zhang, "How much computing capability is enough to run a cloud radio access network?," *IEEE Communications Letters*, vol. 21, pp. 104–107, Jan 2017.

[33] R. S. Alhumaima and H. S. Al-Raweshidy, "Evaluating the energy efficiency of software defined-based cloud radio access networks," *IET Communications*, vol. 10, no. 8, pp. 987–994, 2016.

[34] B. Stern, "Qoriq qonverge portfolio next-generation wireless network bandwidth and capacity enabled by heterogeneous and distributed networks."

[35] G. Auer, V. Giannini, C. Desset, I. Godor, P. Skillermark, M. Olsson, M. A. Imran, D. Sabella, M. J. Gonzalez, O. Blume, *et al.*, "How much energy is needed to run a wireless network," *Wireless Communications, IEEE*, vol. 18, no. 5, pp. 40–49, 2011.

[36] B. Błaszczyszyn, M. K. Karray, and H. P. Keeler, "Using poisson processes to model lattice cellular networks," in *2013 Proceedings IEEE INFOCOM*, pp. 773–781, April 2013.

[37] A. Guo and M. Haenggi, "Spatial stochastic models and metrics for the structure of base stations in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 12, pp. 5800–5812, November 2013.

[38] S. Yang, M. Wang, *et al.*, "A quantum particle swarm optimization," in *Evolutionary Computation, 2004. CEC2004. Congress on*, vol. 1, pp. 320–324, IEEE, 2004.

[39] L. D. De Oliveira, F. Ciriaco, T. Abrao, and P. J. E. Jeszensky, "Particle swarm and quantum particle swarm optimization applied to ds/cdma multiuser detection in flat rayleigh channels," in *Spread Spectrum Techniques and Applications, 2006 IEEE Ninth International Symposium on*, pp. 133–137, IEEE, 2006.

[40] L. Ye, C. Gniady, and J. H. Hartman, "Energy-efficient memory management in virtual machine environments," in *Green Computing Conference and Workshops (IGCC), 2011 International*, pp. 1–8, IEEE, 2011.

[41] M. Portolani and C. Elsen, "Network consolidation for virtualized servers," Sept. 27 2011. US Patent 8,027,354.

[42] L. Minas and B. Ellison, "The problem of power consumption in servers," *Intel Corporation. Dr. Dobb's*, 2009.

[43] R. S. Alhumaima and H. S. Al-Raweshidy, "Modelling the power consumption and trade-offs of virtualised cloud radio access networks," *IET Communications*, vol. 11, no. 7, pp. 1158–1164, 2017.

[44] H. Zhuang, D. Shmelkin, Z. Luo, M. Pikhletsky, and F. Khafizov, "Dynamic spectrum management for intercell interference coordination in lte networks based on traffic patterns," *IEEE Transactions on Vehicular Technology*, vol. 62, pp. 1924–1934, Jun 2013.