Optimization of Virtualization Cost, Processing Power and Network Load of 5G Software-Defined Data Centers

Nazih Salhab, Rana Rahim, Rami Langar

Optimization of Virtualization Cost, Processing Power and Network Load of 5G Software-Defined Data Centers

Nazih Salhab, Rana Rahim, and Rami Langar

***Abstract*—Virtualization is getting unprecedented attention from Mobile Network Operators (MNOs) as it provides agility in deployment, especially when coupled with the Cloud that offers inherent elasticity and load-balancing of resources. MNOs have to ensure operational excellence by meeting several objectives. In this context, we propose in this paper, a framework for optimizing the mapping of next Generation Node-Bs (gNBs) to Software-Defined 5G Core (5GC) delay tolerant Network Functions (NFs). These NFs are considered to be deployed as a Virtual Machine (VM) pool, or containers, in order to minimize cloud computing cost, processing power and at the same time maximize network load. First, we formulate this problem as an integer linear program, while taking into account multiple constraints including Virtual Central Processing Unit (vCPU) capacity, central processing load limits and integrality of mapping relations between gNBs and 5GC NFs. Then, we propose an algorithm to solve large problem instances based on Branch, Cut and Price (BCP) combining all of "Branch and Price", "Branch and Cut" and "Branch and Bound" frameworks. We present several schemes reflecting different optimization goals that the MNO can foster: virtualization cost, power minimization, network load or all. Simulation results demonstrate the good performance of our proposed algorithm to solve the gNBs-VM pool mapping for all evaluated schemes, while also emphasizing the advantages of a particular one (EWoS-333 for Equal Weight optimization Scheme) that can decrease virtualization cost by almost one order of magnitude compared to a static selection scheme, while considering the other two objectives.**

***Index Terms*—Multi-objective optimization, Branch, Cut and Price (BCP), 5G, Mobile Network Operator (MNO), Virtualized Network Functions.**

I. Introduction

Costs breakdown for Mobile Network Operators (MNOs) shows that a large portion of their direct expenses is accounted as cost of goods to deploy their infrastructure as part of capital expenditures. Another significant part is accounted to some indirect cost resulting from operational expenditures of their network, and particularly energy costs.

On top of that, external driving forces stipulate the necessity to abide by green initiatives to minimize power consumption and Carbon Dioxide emissions in data-centers. Accordingly, MNOs have to cut costs, minimize power and maximize their efficiency. Interesting transformation tactics include sharing some of the network's infrastructure and leveraging Software-Defined Infrastructures (SDIs) by deploying network functions on software-defined service centers [1].

Cloud Computing (CC) is gaining momentum among MNOs [2, 3]. Cloud Computing (CC) is getting popularity among Telephone Companies (telcos). For instance, American Telephone and Telegraph (AT&T), the world's largest provider of mobile telephone services, announced that it is becoming a 'public cloud first' company by migrating its workloads to Microsoft public cloud by 2024. They advocate that the clouds allow them to focus on core network capabilities, accelerate their innovation cycle, and empower their workforce while optimizing costs [4]. Furthermore, a leading research and consulting business mandates that in order to be able to compete in the digital world, the adoption of public cloud by telcos is inevitable [5]. They also predicted that telcos will be one of the fastest-growing users of public cloud computing in 2020 as they look to accelerate their new service delivery plan [5]. Cloud Service Providers (CSPs) provide worldwide CC services. Usage of CC allows MNOs to move faster, focus on their business, minimize their hardware footprints, and keep pace with increasing demands in terms of resources. However, costs have to be maintained to a minimum level to maximize profitability [6]. One way to achieve this goal is by exploiting Software-Defined Data Centers (SD-DCs) [6]. Not only, does SD-DC allow to cut costs, but also, it serves as a driver for new business models, provided that it satisfies the latency, bandwidth and distance constraints. MNOs can provision cloud Compute services to implement SDIs of Fifth Generation of Core Network (5GC) virtualized Network Functions (NFs), using Virtual Machines (VMs) offering Virtual Central Processing Units (vCPUs) and Virtual Memory expressed in Gigabytes (GB).

The 5G architecture employs control and user plane separation to have user plane functions and control plane functions and interconnects the Radio Access Network (RAN) to the 5GC through the transport network with next generation interfaces (N1-N4) as depicted in Fig 1. The minimum requirements for control plane latency is 20 ms [7]. Futhermore, according to the requirement R48 of the Next Generation Mo-

Manuscript received on August 13, 2019; revised on January 24, 2020; accepted on April 22, 2020

N. Salhab and R. Langar are with LIGM CNRS-UMR 8049, University Gustave Eiffel, Champs-sur-Marne 77420, France, (E-mails: {nazih.salhab, rami.langar}@univ-eiffel.fr)

R. Rahim is with LTRM, Faculty of Science, Lebanese University, Tripoli, Lebanon, (E-mail: rana.rahim@ul.edu.lb)

bile Networks (NGMN) alliance [8], the maximum guarantee of end-to-end latency of 10 milliseconds is fine for most critical applications such as voice over IP and video over IP.

5GC is envisioned to be built using NF-based approach (Access and Mobility Function (AMF), Session Management Function (SMF), and so on) in a Cloud-native architecture [9] and deployed using a service-based design [10]. Such reference architecture for 5G is backed-up by major standards developing organizations [10–12]. However, MNOs have to dynamically provision their resources to meet several objectives. Knowing that an activation of a VM implementing a 5GC NF is atomic (either active or inactive) and the mapping of new Generation Node-Bs (gNBs) to a VM pool is also binary (connected or not connected), we observe the constraints. A VM pool is a group of VMs that are all clones of the same template and that can be used on demand by any user in a given group. Same concept applies to microservices, where a service is deployed as a container, and pooled over multiple replicas. Pooling Network Functions in the core is not new, it was seen in 4G Evolved Packet Core (EPC) with the Mobility Management Entity (MME-in-pool), and the Serving Gateway (SGw-in-pool) and also in 5G [10].

In this context, we address, in this paper, the problem of dynamic mapping of gNBs to a VM pool in 5G implementing a 5G network function, for example Network Data Analytics Function (NWDAF) for offline, delay tolerant, batch processing, while minimizing the CC cost, the processing power and at the same time maximizing the network load. We refer to this problem as Multi-Performance objective Data Center (MPDC). We first formulate the mapping problem as an Integer Linear Program (ILP). Then, we propose an algorithm to solve it based on Branch, Cut and Price (BCP) combining all of "Branch and Price" (BP), "Branch and Cut" (BC) and "Branch and Bound" (BB) frameworks. We analyze several gNBs clustering strategies and provide guidelines that could be used by MNOs to help them decide on their preferred strategy according to their ultimate goal. We show that an adequate selection of parameters allows reaching multiple objectives and specifically, the Equal Weight optimization Scheme (EWoS-333) provides gains on different levels (power, CC cost and network load) and particularly, performs very close to a pure cost minimization scheme, while addressing power and throughput objectives.
In summary, our key contributions are the following:

- We formulate the dynamic mapping of gNBs and VM pools in the 5G context as an Integer Linear Program (ILP), while minimizing CC costs, processing power and maximizing the grouping of low-loaded gNBs to reduce the complexity and the power consumption.
- We propose an algorithm for solving this problem using BCP framework, which is a combination of the Branch-and-Bound, Branch-and-cut and Column Generation methods for efficiently solving large-scale ILP problems.
- We validate proposed algorithm by means of simulation and show the effectiveness of our proposal compared to other solutions using pricing data extracted from Google Cloud Platform (GCP) price-list [13].

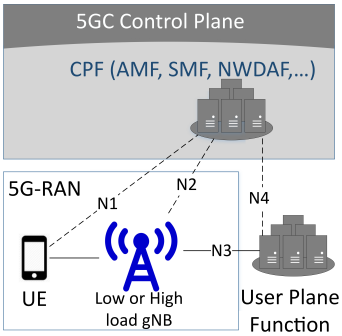

Fig. 1: Simplified 5G Architecture

The remainder of this paper is organized as follows. In section II, we present an overview of related work. Section III describes the system model and formulates the problem as a generic weighted optimization one. Section IV details the proposed algorithm. We elaborate the performance of our proposal and discuss the obtained results in section V. Finally, we conclude our paper and provide some perspectives in Section VI .

II. Related Work

Optimizing cost, power and load for a SDIs has triggered considerable interest among researchers in the past few years. In what follows, we discuss a selection of pertaining papers classified by research areas.

A. Cloud Computing Cost minimization based on storage

Authors in [14] proposed a cost-based placement of virtualized deep packet inspection function in Network Function Virtualization (NFV) infrastructures using ILP formulation. They expressed the placement optimization as a cost minimization problem. They presented the situation as multi-commodity flow problem and solved it using a centrality-based greedy algorithm that is based on graph theory.
Authors in [15] proposed a solution, namely DAR for Data storage, request Allocation and resource Reservation, minimizing the cost, while meeting multiple Service Level Objectives (SLOs) across multiple CSPs. They modeled the cost minimization problem under SLO constraints using Integer Programming and proposed a dominant cost based data allocation algorithm and an optimal resource reservation algorithm. They formulated the SLO as constraints not as objectives.
In [16], the authors propose SPANStore (Storage Provider Aggregating Networked Store) to minimize cloud storage costs, while answering latency and availability objectives across multiple CSPs. They combined three objectives to minimize costs consisting of i) increasing distribution of data centers exploiting pricing differences among CSPs, ii) fetching workload metrics at adequate granularity to trade-off replication

versus latency while meeting fault-tolerance and consistency constraints and iii) minimizing resources' usage during two-phase locking and data propagation. It is worth noting that all of these papers [14–16] treated the cost optimization by minimizing "Storage" resources unlike our approach, which targets minimization of "Compute" resources, making it more suitable to stateless applications.

B. Cloud Computing cost saving through prediction

In [17], the authors modeled the VM deployment and classified them as reserved and on-demand instances to optimize their CC costs in an IaaS. They proposed a strategy that is based on large deviation principle which calculates a number of VMs responding to demands taking overload probability into consideration. Also, aiming to further reduce the total cost, authors of [17], added a dynamic approach to predict the load using auto-regressive model calculating the number of instances to be reserved for the computation requirements. In [18], CC costs saving were achieved by exploiting the discounts resulting from scheduling reservation of resources on recurring basis in advance. These two latter approaches rely on prediction to save costs as opposed to our approach, where we propose a dynamic mapping of gNBs and VM pools based on a generic weighted optimization problem.

C. Savings as results of virtualization

Authors in [19] addressed a virtualization scheme of VM pool aiming to minimize power consumption constrained by processing capacity. They used a heuristic that is based on simulated annealing that is a well-known method for solving unconstrained and bound-constrained optimization allowing getting near-optimal result at minimum time.
Authors in [20] focused on the setup of service chains driven by the emergence of NFV. They proposed a directed acyclic graph to map chain topology of services onto a physical path between source and sink nodes. However, the dynamic link optimized placement was not considered.
Authors in [21] designed a multi-tier ecosystem of stream applications. They modeled the energy of the target ecosystem by accounting the virtualized and multi-core nature of the Fog/cloud servers. They approached such problem using gradient-based adaptive iterations and genetic algorithms.

Authors in [22] discussed a joint optimization of data-center selection and video streaming rendering in a geo-distributed Cloud platform. They proposed an online algorithm to save operational costs by dynamically choosing right data centers for both broadcasters and viewers at the same time.

D. Optimized placement of Network Functions

Authors in [23] proposed an online algorithm for dynamic Software-Defined Network (SDN) controller assignment in Data center networks aiming to minimize total cost caused by response time and maintenance on the cluster of controllers. They considered only one objective without considering additional competing objectives.
Authors in [24] proposed a scalable resource allocation scheme, namely, ClusPR that addresses multiple objectives on NFs which are clusters formation, placement of NFs and routing of related flows. They modeled such problem as an ILP aiming to find end-to-end route of flows, while maintaining the precedence constraint among such NFs of a service chain. They proposed two algorithms for offline and online processing of such resource allocation problem aiming to minimize path stretch and NFs load and maximize overall network utilization.
Authors in [25] proposed three optimization models aiming to address cost minimization of Core Network virtualization in 5G data center based on SDN and NFV. They found out a trade-off between centralized and distributed data centers deployments. They proposed a pareto optimal multi-objective model that balances network and data center cost.
Authors in [26] presented a cloud resource allocation problem targeting cost minimization and quality of service maximization when deploying applications in the cloud. They evaluated the performance of their proposed algorithm using pricing data from Amazon web service [27] and Rackspace [28].

Authors in [29] proposed a dynamic energy-saving model with NFV using an M/M/c queuing network to increase the utilization of the machines. They formulated an energy-cost optimization problem with capacity and delay constraints.

E. Generalized assignments heuristics and exact solutions

Authors in [30] considered cost minimization of a scheduling problem and proposed a heuristic using BB named NMSP (Node Migration Scheduling Problem). They conducted a simulation on 4G network consisting of 40 nodes and found-out that optimality is guaranteed for small networks of up to 40 nodes.

Authors in [31] considered assignment problems and proposed a heuristic named MGAP (Multilevel Generalized Assignment Problem) using BC to minimize assignment cost of jobs to machines with capacity constraint.

Authors in [32] proposed a framework for the BBU selection based on resiliency and price using BP framework that relies on Column Generation (CG) and BB.

Authors in [33] studied the generic Branch, Cut and Price approach from analytical point of view and proved its ability in providing exact solutions in large scale problems.

Based on these works, we propose here a new approach for mapping dynamically gNBs to a VM pool in the 5G context, while minimizing the CC cost, the processing power and at the same time maximizing the network load. Our approach relies on a combination of the Branch-and-Bound [30], Branch-and-Cut [31] and Column Generation (CG) [34] methods for efficiently solving large problem instances. In addition, we conduct simulations using real up-to-date pricing data from GCP [13] and show the effectiveness of our proposal compared to alternative solutions.

III. System model and problem formulation

A. System Model

We consider a 5G simplified architecture consisting of three components, which are: gNBs, VM pools (or containers) and a backhaul transport network for interconnecting the gNBs to the 5GC [35]. We assume that the latency imposed by hosting the 5GC delay tolerant services on the Cloud is acceptable when backhauling the gNBs. We will verify this assumption in the performance evaluation (Section V). We consider an area hierarchy that is composed of a cluster of S gNBs. The VM pool is denoted by $\mathbb{V}$ with i as index such that $\mathbb{V} = \{i | 1 \leq i \leq N\}$. The set of gNBs is denoted by $\mathbb{G}$ with j as index such that $\mathbb{G} = \{j | 1 \leq j \leq S\}$. We define a binary decision variable denoted by r_{ij} to express the VM pool i to gNB j mapping relation. Accordingly, r_{ij} is equal to 1 if gNB j is mapped to VM pool i and 0, otherwise. The average utilization u_i of a VM pool i is expressed as follows:

$$u_i = \frac{1}{L_i} \sum_{j=1}^{S} r_{ij}.l_j \tag{1}$$

where l_j denotes the traffic load on the gNB j and L_i denotes the maximum capacity of the VM pool i. To account for the backhaul load of the different gNBs and their impact of the VM pool, knowing that each VM pool i has a limited capacity L_i, we consider that this maximum throughput imposes a Central Processing (CP) load to the VM. To take this constraint into consideration, we define parameter m_i to denote the CP load on VMs resulting from each gNB according to its load and stipulate that a maximum CP load of B expressed in percent. On the other hand, we denote by C_i the cost of instantiating a VM i to implement a 5G Service. To normalize the costs, we denote by $max(C)$ the maximum value of the VMs costs in the CSP pricing list. We consider two sets of gNBs, formed according to the traffic load of these gNBs [36]. We denote them as $\mathbb{S}_\mathbb{L}$ for low-loaded set of gNBs and $\mathbb{S}_\mathbb{H}$ for high-loaded set of gNBs. We propose this segregation of gNBs based on traffic load, since asymmetrical traffic between day and night in addition to differences among business and residential areas in term of processing capacity requirement for services are observed. For the sake of clarity, we define a binary decision variable x_i as summation of r_{ij} over j as follows.

$$x_i = \sum_{j=1}^{S} r_{ij} \tag{2}$$

This latter variable expresses the active state of a $\mathrm{VM_i}$ such that x_i is 0 when $\sum_{j=1}^{\mathbb{S}} r_{ij} = 0$, meaning that not a single $\mathrm{gNB_j}$ is mapped to a $\mathrm{VM_i}$. We denote by P_i the consumed power and we model it as a function of the VM average utilization u_i with a linearity slope of $\lambda.\mathrm{P_{max}}$, provided that $\mathrm{P_{max}}$ denotes the maximum consumed power by the VM when fully loaded. Coefficient λ is normalized to have values between 0 and 1. P_0 denotes the consumed power during idle mode, that is the residual power when there is no utilization (u_i=0). Accordingly, P_i is expressed as follows.

$$P_i = P_0 + \lambda.\mathrm{P_{max}}.u_i \tag{3}$$

Tables I and II report the notations used in our system model as decision variables and parameters.

B. Problem Formulation

We formulate our gNBs-VM pool Mapping Problem for SD-DC (P) as an Integer Linear Program (ILP) with the following generic weighted objective function composed of three homogenized terms.

$$\text{(P)} \min_{r} \quad \alpha \sum_{i=1}^{N} \frac{x_i.P_0 + \lambda P_{max} u_i}{P_{max}} + \beta \sum_{i=1}^{N} x_i \frac{C_i}{max(C)} - \gamma \sum_{i=1}^{N} \frac{\sum_{j \in \mathbb{S}_\mathbb{L}} r_{ij} l_j}{\sum_{j \in \mathbb{S}_\mathbb{L}} l_j} \tag{4a}$$

s.t.

$$\sum_{i=1}^{N} \sum_{j \in \mathbb{S}_\mathbb{H}} r_{ij} l_j = \sum_{j \in \mathbb{S}_\mathbb{H}} l_j \tag{4b}$$

$$\sum_{j=1}^{S} r_{ij} l_j \leq L_i, \forall i \in \{1, \ldots, N\} \tag{4c}$$

$$\sum_{i=1}^{N} r_{ij} \leq 1, \forall j \in \{1, \ldots, S\} \tag{4d}$$

$$\sum_{i=1}^{N} m_i \sum_{j \in \mathbb{S}_\mathbb{H}} r_{ij} l_j \leq B, \forall i \in \{1, \ldots, N\} \tag{4e}$$

$$x_i = \sum_{j=1}^{S} r_{ij}, \forall i \in \{1, \ldots, N\} \tag{4f}$$

$$x_i \in \{0, 1\}, \forall i \in \{1, \ldots, N\} \tag{4g}$$

$$r_{ij} \in \{0, 1\}, \forall i \in \{1, \ldots, N\}, j \in \{1, \ldots, S\} \tag{4h}$$

The proposed objective function in (4a) consists of minimizing the total VM pool power consumption, in addition to minimizing CC cost, while maximizing the traffic load that could be processed by the VM pool from the low-loaded gNBs. The three unit-less weights denoted as α, β and γ are the coefficients of these normalized objective terms with values ranging between 0 and 1 and having a sum equal to 1 ($\alpha + \beta + \gamma = 1$). We assume that these parameters are set by the MNOs to reflect their optimization strategy according to a choice of prevailing factors (prioritization of Power minimization over CC cost or load maximization from low-loaded gNBs).

Constraint (4b) specifies that the traffic of high-loaded gNBs is exactly handled in total by the VM pool.

Constraint (4c) ensures that the sum of load of gNBs associated to a VM pool does not exceed the capacity (L_i) of such VM pool.

Constraint (4d) stipulates that no gNB could be associated to

TABLE I: Decision Variables Notations

Variable	Description
r_{ij}	decision variable of mapping gNB j to VM pool i
v_{ij}	extreme point: result of transformation of r_ij
w_{ij}	extreme ray: result of transformation of r_ij
x_i	Active state of VM i
z_k^i	decision variable of feasibility of selected solution
$\dot{z}_k^i$	First dimension of z_k^i
$\ddot{z}_k^i$	Second dimension of z_k^i

more that one VM pool. Indeed, having the summation of all r_{ij} binary variables less or equal to one is only satisfied if at most one particular $r_{i_0 j_0}$ is equal to one.

Constraint (4e) stipulates that the CP load imposed by the formed gNBs cluster associated to a VM pool do not exceed the maximum allowed CP load B. The parameter m_i represents the accrued amount in term of CP load percentage per served load in Mbps.

Constraint (4f) specifies the relation between the two decisions variables x_i and r_{ij} as per the definition of x_i itself as elaborated in equation (2).

Constraints (4g) and (4h) stipulate that the decision variables are binary as a VM pool can only be active or inactive (x_i=0 or 1) and an gNB j can be either associated to a VM pool i or not (r_{ij}=0 or 1).

IV. Proposed BCP Algorithm

Our multi-objective mapping problem formalized in (P) is an ILP and cannot be solved directly using convex optimization techniques. Integrality constraints (4g), (4h) make it harder to solve compared to Linear Programs (LPs) where, in the latter, the decision variables can take any arbitrary real value. Problem (P) is NP-hard [24, 26, 32]. A naive method to generate optimum solution, is by exhaustively evaluating all N^S possible combinations of gNB-VM pool assignments. However, it is impractical for large-scale networks as the computation time, for such approach, increases exponentially with the number of gNBs. To find a solution to our problem, we propose an algorithm based on the Branch, Cut and Price (BCP) framework which combines column generation, cuts and branch-and-bound approaches to find a optimal solution at minimum time. After relaxing the integrality constraints at a first stage, BCP algorithm consists in using Column Generation to progressively and dynamically generating promising solutions for the Master Problem detailed here-after. We then use valid inequalities to cut feasible region and to strengthen the linear relaxation so that a solution come closer to integers. We finally use branch-and-bound to systematically search for the solution. Column generation iteration consists in solving the linear relaxation of the Master Problem and deciding which column will be added. In such case, the ILP is simplified to an LP. Accordingly, a sub-problem called the "Pricing Problem" is created to identify which columns should enter the basis in order to increase the objective function. If such columns are found, the LP is then re-optimized. In column generation, each iteration consists of (1) optimizing a restricted master problem (RMP) to determine current optimal objective function and (2) finding a variable with reduced cost influencing the behavior of the dual variables. Cuts are constraints that are dynamically added to our model to restrict non integer solutions. Using cuts, we eliminate a non-integer solution that results from the linear relaxation. We detail, next, the steps to have the Master and Pricing problems for the proposed BCP algorithm.

A. Problem Reformulation

Based on the structure of our original problem (P) and using Minkowski-Weyl's representation theorem [37] that states that every polyhedron $\mathbb{P}$ can be represented in the form of a convex linear expression of extreme points and extreme rays of such polyhedron, we transform our original problem using $\mathbb{P} = \{r \in \mathbb{R}^n : r = \sum \rho.v + \sum \mu.w\}$ where v are the extreme points, w are the extreme rays and ρ, μ are linear coefficients. We use two binary variables v_{ij} and w_{ij} instead of the initial decision variable r_{ij} for the low-traffic load l_j^L and high-traffic load l_j^H gNBs assignments, respectively. Same definition remains for x_i, as $x_i = 0$ if VM_i is inactive ($\sum_{j\in\mathbb{S}_\mathbb{L}} v_{ij} + \sum_{j\in\mathbb{S}_\mathbb{H}} w_{ij} = 0$) and $x_i = 1$ otherwise. Developing and simplifying our original mapping problem (P), we get our Transformed Problem (TP), as follows.

$$\text{(TP)} \min_{v,w} \quad \sum_{i=1}^{N} \Phi_i.x_i + \sum_{i=1}^{N}\sum_{j\in\mathbb{S}_\mathbb{L}} \Omega_i v_{ij} l_j^L + \sum_{i=1}^{N}\sum_{j\in\mathbb{S}_\mathbb{H}} \Psi_i w_{ij} l_j^H \tag{5a}$$

s.t.

$$\sum_{i=1}^{N}\sum_{j\in\mathbb{S}_\mathbb{H}} w_{ij} l_j^H = \sum_{j\in\mathbb{S}_\mathbb{H}} l_j^H \tag{5b}$$

$$\sum_{j\in\mathbb{S}_\mathbb{L}} v_{ij}.l_j^L + \sum_{j\in\mathbb{S}_\mathbb{H}} w_{ij}.l_j^H \le L_i,\ \forall i \in \{1,\dots,N\} \tag{5c}$$

$$\sum_{i=1}^{N} v_{ij} \le 1,\ \forall j \in \mathbb{S}_\mathbb{L} \tag{5d}$$

$$\sum_{i=1}^{N} w_{ij} \le 1,\ \forall j \in \mathbb{S}_\mathbb{H} \tag{5e}$$

$$\sum_{i=1}^{N} m_i\Big(\sum_{j\in\mathbb{S}_\mathbb{L}} v_{ij} l_j^L + \sum_{j\in\mathbb{S}_\mathbb{H}} w_{ij} l_j^H\Big) \le B \tag{5f}$$

$$x_i = \sum_{j\in\mathbb{S}_\mathbb{L}} v_{ij} + \sum_{j\in\mathbb{S}_\mathbb{H}} w_{ij},\ \forall i \in \{1,\dots,N\} \tag{5g}$$

$$v_{ij} \in \{0,1\},\ \forall i \in \{1,\dots,N\}, \forall j \in \mathbb{S}_\mathbb{L} \tag{5h}$$

$$w_{ij} \in \{0,1\},\ \forall i \in \{1,\dots,N\}, \forall j \in \mathbb{S}_\mathbb{H} \tag{5i}$$

where $\Phi_i = \frac{\alpha P_0}{P_{max}} + \beta\frac{C_i}{max(C)}$, $\Omega_i = \frac{\alpha\lambda}{L_i} - \frac{\gamma}{\sum_{j\in\mathbb{S}_\mathbb{L}} l_j^L}$ and $\Psi_i = \frac{\alpha\lambda}{L_i}$. Let the two sets of feasible possible assignments of low and high-loaded gNBs to VM pool i be $\Xi_i^L = \{v_1^i, v_2^i, \dots, v_{k_i}^i\}$ and $\Xi_i^H = \{w_1^i, w_2^i, \dots, w_{k_i}^i\}$. Two particular variables of Ξ_i^L and Ξ_i^H, $v_k^i = \{v_{1k}^i, v_{2k}^i, \dots, v_{Sk}^i\}$ and $w_k^i = \{w_{1k}^i, w_{2k}^i, \dots, w_{Sk}^i\}$ are a valid solution to our transformed problem formulated in (5a). According to Dantzig-Wolfe's decomposition [38] that

TABLE II: Parameters Notations

Parameter	Description
α	coefficient of processing power
β	coefficient of virtualization cost
γ	coefficient of low-loaded gNB
λ	normalization coefficient
B	Maximum allowed CP Load
C_i	CC cost of VM i
L_i	Max capacity of VM i
l_j^H	Load of j-th gNB with high-load
l_j^L	Load of j-th gNB with low-load
k_i	count of feasible points
m_i	Delta CP load percentage per Mbps
P_{max}	Maximum Power consumed by VM
P_0	Idle Residual Power of VM
u_i	Utilization of VM i
Max(c)	Maximum cost of instantiating VM
N	Maximum Number of VM
S	Maximum Number of gNB
$\mathbb{S}_H$	Set of high-loaded gNBs
$\mathbb{S}_L$	Set of low-loaded gNBs

sub-divides the problem into a Master and Pricing Problem, we define a new variable $z_k^i = (\dot{z}_k^i, \ddot{z}_k^i)$ as a two-dimensions decision variable, that reflects the feasibility of the selected solution. Accordingly, z_k^i = (1,1) when (v_k^i, w_k^i) is feasible and (0,0) othwerwise. The count of feasible points is denoted by k_i. The Master Problem (MP) is a sub-version of the previous TP, where we disregard the complicating (coupling) constraints (5c). We express, in the following, our Master Problem (MP).

$$\text{(MP)} \min_{z} \quad \sum_{k=1}^{k_i}\sum_{i=1}^{N}(\Phi_i x_i + \Omega_i \sum_{j\in\mathbb{S}_L} v_{ij} l_j^L \dot{z}_k^i + \Psi_i \sum_{j\in\mathbb{S}_H} w_{ij} l_j^H \ddot{z}_k^i) \tag{6a}$$

s.t.

$$\sum_{k=1}^{k_i}\sum_{i=1}^{N}\sum_{j\in\mathbb{S}_H} \ddot{z}_k^i w_{jk}^i l_j^H = \sum_{j\in\mathbb{S}_H} l_j^H \tag{6b}$$

$$\sum_{k=1}^{k_i} \dot{z}_k^i \le 1,\ \forall i \in \{1,\dots,N\} \tag{6c}$$

$$\sum_{k=1}^{k_i} \ddot{z}_k^i \le 1,\ \forall i \in \{1,\dots,N\} \tag{6d}$$

$$\sum_{k=1}^{k_i}\sum_{i=1}^{N} \dot{z}_k^i v_{ij} \le 1,\ \forall j \in \mathbb{S}_L \tag{6e}$$

$$\sum_{k=1}^{k_i}\sum_{i=1}^{N} \ddot{z}_k^i w_{ij} \le 1,\ \forall j \in \mathbb{S}_H \tag{6f}$$

$$x_i = \sum_{j\in\mathbb{S}_L} \dot{z}_k^i v_{ij} + \sum_{j\in\mathbb{S}_H} \ddot{z}_k^i w_{ij},\ \forall i \in \{1,\dots,N\} \tag{6g}$$

$$\dot{z}_k^i \in \{0,1\},\ \forall i \in \{1,\dots,N\}, k \in \{1,\dots,k_i\} \tag{6h}$$

$$\ddot{z}_k^i \in \{0,1\},\ \forall i \in \{1,\dots,N\}, k \in \{1,\dots,k_i\} \tag{6i}$$

In the MP, z_k^i represents a feasible assignment of gNBs to a VM i. Note that this decomposition is performed to obtain a problem formulation that gives better bounds compared to when the relaxation of the original formulation is solved. However, as we get many variables, the MP cannot be solved directly due to its big number of columns. Accordingly, We define our Restricted Master Problem (RMP) that considers a subset of the columns to be solved. In the RMP, the values of variables that do not figure in the equations are padded as zero. For the RMP, we consider z^* as the corresponding dual solution. We add a number of columns with positive reduced cost that results from solving following sub-problems:

$$\min_{1\le i\le N} \{o^i - z^{*i}\} \tag{7}$$

where $o^i = (\dot{o}^i, \ddot{o}^i)$ is the optimal solution of our Pricing Problem (PP), that is expressed as follows.

$$\text{(PP)} \min_{v,w} \quad \Phi_i x_i + \Omega_i \sum_{j\in\mathbb{S}_L} v_j^i (l_j^L - v_j^*) + \Psi_i \sum_{j\in\mathbb{S}_H} w_j^i (l_j^H - w_j^*) \tag{8a}$$

s.t.

$$\sum_{j\in\mathbb{S}_L} v_j^i l_j^L + \sum_{j\in\mathbb{S}_H} w_j^i l_j^H \le L_i,\ \forall i \in \{1,\dots,N\} \tag{8b}$$

$$v_{ij} \in \{0,1\},\ \forall i \in \{1,\dots,N\}, \forall j \in \{1,\dots,S\} \tag{8c}$$

$$w_{ij} \in \{0,1\},\ \forall i \in \{1,\dots,N\}, \forall j \in \{1,\dots,S\} \tag{8d}$$

The two values v_j^* and w_j^* correspond to the optimal dual price resulting from solving the RMP associated with the partitioning constraints of low and high-loaded gNB_j. Note that, by solving PP, we get the optimal gNBs to VM pool i associations. Solution to problem (7) would be the gNB-VM pool mapping with minimum objective function.

B. Proposed BCP algorithm to solve the MPDC problem

To find a solution to our original problem (P), we propose the code listed in Algorithm 1. We start by solving the reference Linear Program with relaxed constraints to get the ideal lower bound solution. Then, we solve the PP and RMP and we begin the column generation process by evaluating new nodes to enter the basis of RMP if they provide reduced value. Then, we proceed to cut generation if some coefficients are not integer to enforce integrality on next run. Finally, we branch and update the list of unprocessed nodes. The processing keeps on iterating as long as the stop criterion is not reached. Such stop criterion could be either a time-limit or a relative gap tolerance between the found value and lower bound value.

Note that the BCP is a combination of all of the branch-and-cut, branch-and-price and branch-and-bound, and it is known to be an exact algorithm providing the optimal solution, as proved in [33]. Indeed, the first two stages (Column generation and cuts) consist of shrinking the bounds of the interval containing the optimal solution. The last stage of the BCP is the Branch and Bound, and acts similarly to a brute force search with some intelligence. Specifically, in the Branch and Bound, we keep splitting the search space, and work on the sub-problems. The bounding part of the algorithm stop us from

exploring a particular branch only if it is confirmed that it does not contain the optimal solution. Since, we do not discard any potential global optimal, Branch and Bound will find one if it exists.

Algorithm 1: BCP-based MPDC Listing

Data: Objective function and constraints
Result: Optimum feasible solution
Initialize Problem (P);
Solve LP with relaxed constraints;
Get Lower-Bound (LB) solution;
(A) Choose a new node;
(B) Solve Restricted Master Problem (RMP);
Evaluate a new node;
if *(reduced value found)* **then**
  Add such column to the basis of RMP;
end
Solve PP to optimality;
if *(solution with reduced value found)* **then**
  Add to RMP;
  goto (B);
end
if *(no solution with negative reduced value is found)* **then**
  update lower bound;
end
if *(∃ LB of other branch < computed LB)* **then**
  remove this node;
  goto (A);
end
if *(integer coefficient is not met)* **then**
  Generate cuts;
  Add them to the RMP;
  goto (B);
end
if *(Solution is integral)* **then**
  Update upper bound;
else
  branch and add children nodes to unprocessed;
end
if *stop criterion is reached* **then**
  quit;
end
goto (A);

V. Performance Evaluation

In this section, we quantify the benefits of the proposed BCP algorithm to solve our MPDC mapping problem, while considering different scenarios.

We have chosen to conduct our simulations using GCP pricing data since it offers low latency within the stipulated limit of NGMN on the backhaul. To validate this assumption, we spin the smallest VM instances, called (f1-Micro), to Ubuntu 16.04 on major European regions covered by GCP and tested within each VM the ping for 100 times to other public IPs of instantiated VMs. After averaging, we found that the ping takes less than 10 ms between several point of presence in Europe, as reported in Table III.

TABLE III: Latency in milliseconds for some regions in Europe in GCP

From\To	Belgium	London	Frankfurt	Netherlands
Belgium	N/A	6.1	7.8	107.7
London	6.2	N/A	13.4	10.5
Frankfurt	7.7	12.7	N/A	7.4
Netherlands	107.7	11.9	8.8	N/A

TABLE IV: Optimization strategies

Scheme	Mnemonic	α	β	γ
Power Minimization	PMiS-100	1	0	0
Cost Minimization	CMiS-010	0	1	0
Low-Throughput Maximization	LTMaS-001	0	0	1
Equal Weight Optimization	EWoS-333	1/3	1/3	1/3

We considered different combinations of the parameters α, β and γ, as summarized in Table IV. PMiS-100 initializes α, β and γ as 1,0,0, respectively, aiming to minimize the VM pool processing power only. CMiS-010 initializes α, β and γ as 0,1,0, respectively, aiming to minimize the CC cost only. LTMaS-001 initializes α, β and γ as 0,0,1, respectively, aiminig to maximize the load resulting of handled low-loaded gNBs. In EWoS-333, α, β and γ are initialized as $\frac{1}{3},\frac{1}{3},\frac{1}{3}$, respectively, aiming to meet all these three objectives together. Finally, Static Selection Strategy (SSS) assigns one-to-one VM to gNB mapping statically. This means that each gNB is handled by one VM. We use different values for S and N. Unless specified otherwise, during our simulations, we consider a total number of gNBs (S = 40) and a total number of VMs (N = 8) [30]. As in [32, 39], we considered time-variant traffic profiles for both business and residential gNBs as depicted in Fig. 2.

Business area has its busy hour around noon and is very low starting midnight till early morning. However, residential area has its peaks at night time, which is normal, when workers are back to their homes.

We used commercial solver IBM ILOG CPLEX [40] to solve the optimization problems formulated in Section IV-A.

A. Complexity Analysis

1) Comparison to Exhaustive search: We considered multiple small-scale deployment scenarios according to the number of VMs (N) and number of gNBs (S), as elaborated in Table

TABLE V: Computation time of MPDC using BCP and Exhaustive search

		BCP-based MPDC		Exhaustive Search MPDC	
N	**S**	**Ticks**	**Seconds**	**Ticks**	**Seconds**
8	**40**	44.12	0.08	534	0.83
10	**45**	98.68	0.24	2760.12	6.32
12	**50**	174.32	0.42	19507.24	47
15	**75**	332.51	0.68	108065.75	221
18	**100**	457.98	0.88	304452.61	585
20	**125**	571.23	0.93	405465.12	823
25	**150**	963.69	1.46	807615.61	2018
30	**175**	1256.87	1.83	914825.75	14561
35	**200**	1771.13	2.72	Intractable	Intractable

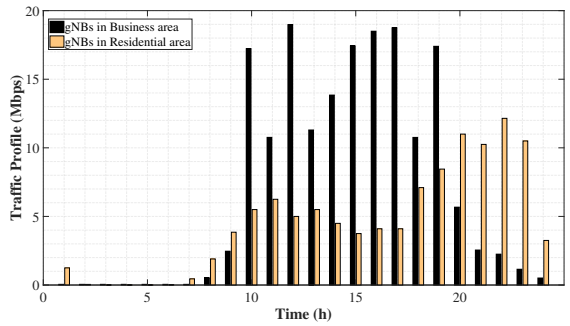

Fig. 2: 24h traffic Profile of gNBs in Business and Residential areas [32, 39]

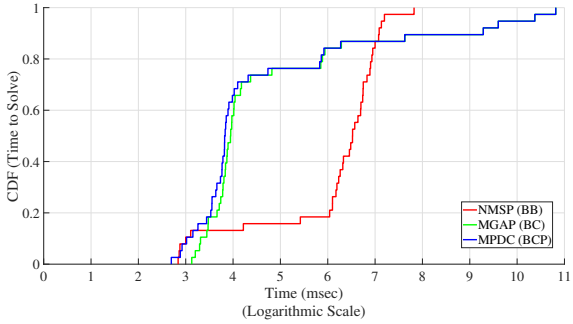

Fig. 4: CDF for NMSP, MGAP and MPDC

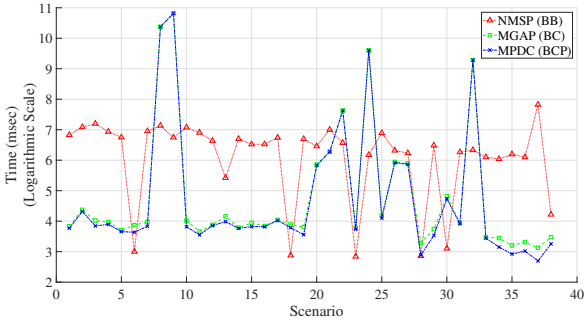

Fig. 3: Time Comparison for NMSP, MGAP and BCP-based MPDC

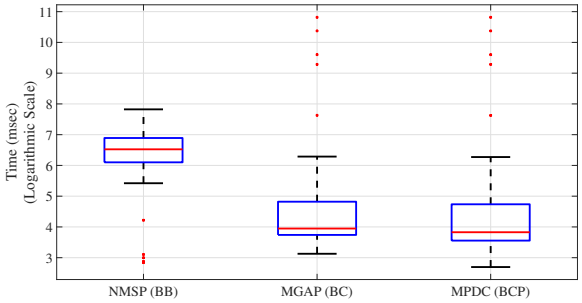

Fig. 5: Boxplot Comparison for NMSP, MGAP and MPDC

V. We measured the performance using two time units: system time measured in seconds and time limits using "ticks" [40] measured deterministically, with up to four threads.

As reported in Table V, we can see that BCP algorithm reduces considerably the computation time compared to the exhaustive search approach. Indeed, using BCP, 0.07 second was needed to solve the gNBs-VM pool mapping problem instead of 0.81 second, that is almost a decrease of one order of magnitude. Starting from (N, S) = (12, 50), we see two orders of magnitude difference in the average time between exhaustive search and a BCP-based approach. For a configuration of (N,S)= (35, 200), exhaustive search becomes intractable, meanwhile computation time using BCP is still in control, and is less than three seconds.

2) Comparison to Branch-and-Bound and Branch-and-Cut: To assess the effectiveness of a BCP-based approach, we compare it with two solutions: NMSP [30] using Branch-and-Bound, and MGAP [31] using Branch-and-Cut. Note that both NMSP and MGAP schemes were first adapted to our context before using them in the comparison. We considered 200 scenarios with variable L_i from 200 to 2190 with a step of 10. In this scenario, in order to control simulation time, we impose a time-limit of 120 seconds. Out of these 200 scenarios, NMSP failed to find an optimal solution within the stipulated time for 139 times out of 200. MGAP failed to find 141 times and BCP-based MPDC only failed 105 times out of 200.

Fig. 3 depicts a comparison of computation time for the three approaches (NMSP, MGAP and MPDC) for the last 40 out of the studied 200 scenarios. On the other hand, we can observe that even with good pricing starting point, at least a call to the exact pricing is necessary to establish a valid dual bound. Note that when the addition of a cut changes the structure of the pricing problem, it makes it harder to solve the sub-problem to optimality, and thus, increases the risk of ending up with an intractable problem. In such case, the whole algorithm would fail as seen in the blue peaks shown in this figure.

Fig. 4 depicts the Cumulative Distribution Function (CDF) of these three approaches to find a solution before the stop criterion. We can see that MGAP and BCP-based MPDC trends are very close. However, MPDC-based approach outperforms MGAP in a lot of scenarios thanks to the usage of cuts on top of column generation and branch-and-bound.
We note that the cuts are dynamically generated by CPLEX. All of the following cuts (Clique cuts, Cover cuts, Flow cuts, Gomory fractional cuts, Implied bound cuts, Mixed integer rounding cuts, and Zero-half cuts) are made available within the solver library and are used during the runtime.

Although, MGAP and BCP-based MPDC are likely to find an optimal solution, we notice that there are some outliers (peaks) where both failed in finding an optimal solution, as depicted in Fig. 5. Such failures are due to the impossibility of solution existence stipulated by the constraints according to the value of L_i.

Table VI reports the average time in ticks and seconds for all of the non-failing scenarios. From that table, we can clearly see that MGAP and BCP-based MPDC are very close, both outperforming the NMSP scheme.

TABLE VI: Computation time of NMSP, MGAP and BCP-based MPDC

NMSP (BB) [30]		**MGAP (BC) [31]**		**MPDC (BCP)**	
ticks	seconds	ticks	seconds	ticks	seconds
418.03	0.69	117.963	0.149	97.45	0.143

TABLE VII: Simulation parameters

Parameter	**Value**
Number of VMs (N)	8
Total Number of gNBs (S) [30]	40
Number of residential gNBs	4
Idle mode Power (P_0) [42]	60 watts
Maximum Power (P_max) [42]	275 watts
λ	1
CP Load B (%) [42]	75
VM capacity L_i (Mbps)	200, 210, .., 2190
Fixed capacity L_0 (Mbps) [43]	225
Cost of 4 VM types ($/hour) [13]	[0.0475 0.0592 0.03545 0.1575]

In what follows, we consider a fixed value of L_i that we denote as L_0 to consider the case of homogeneous VMs (Containers in this case) with constant capacity to be in-line with the microservice architecture best practices [41]. We present the performance comparison of the aforementioned five strategies (i.e., PMiS-100, CMiS-010, LTMaS-001, EWoS-333, and SSS) using different performance metrics:

- VM processing power
- Cloud Computing (CC) cost
- Number of active gNBs and VM pool
- Percentage of handled low-loaded gNBs
- Average number of active VMs
- And average Central Processing (CP) load.

The simulation parameters are reported in Table VII.

B. Total VM pool processing power

Fig. 6 compares the hourly VM pool processing power. All trends are almost inline with the traffic trend of business area gNBs. Obviously, PMiS-100 provides the least power consumption as this is its emphasis. Interestingly, same trend goes for the CMiS-010, where due to power minimization, less VMs are activated and consequently less CC costs are incurred. On the other hand, the SSS scheme is the worst performing strategy as it statically allocates the VMs regardless of the dynamics of the traffic and does not leverage the advantages of statistical multiplexing when pooling resources. EWoS-333 is almost following the same trend, however, it consumes more power at certain peak hours, namely starting from the period 16:00 till midnight. LTMaS-001 has the second highest power consumption as in this strategy, more traffic originated from low-loaded gNBs is handled.

C. Cloud Computing (CC) cost

We present in Fig. 7 the hourly CC cost resulting from the activation of the VM implementing the 5GC service. Apart from the SSS scheme, where the CC cost is flat due to always-on state of VMs, LTMaS-001 is the second top costing strategy although it overlaps with PMiS-100 in early morning before 9:00 and late in the evening starting 21:00. This can

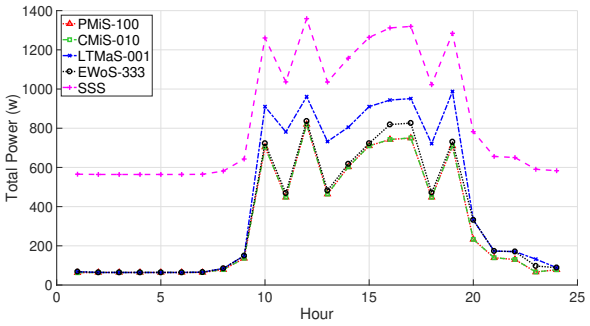

Fig. 6: Total VM Power (w) versus 24-hour time window

TABLE VIII: Daily, Monthly and Yearly Savings ($) for Small Network

Average saving	PMiS_100	CMiS_010	LTMaS_001	EWoS_333
Daily	10.06	12.8	8.28	12.67
Monthly	301.94	383.94	248.27	380.02
Yearly	3623.31	4607.24	2979.29	4560.28

be explained by the fact that this approach handles more cells and thus increases the number of VMs to provide baseband resources for it and consequently increases the CC cost. EWoS-333 and CMiS-010 approaches are the least costing strategies despite the fact that during some hours the EWoS-333 scheme costs a little bit more than CMiS-010, precisely at hours 16:00, 17:00 and 20:00.

This is clearly shown in Fig. 8, where the maximum saving are insured by CMiS-010 followed by EWoS-333 with minor decreases at those hours.

Table VIII quantifies the savings with respect to SSS. Daily savings are computed as the difference between CC costs of each strategy and SSS. Aggregation to month and year is done by multiplying by 30 days per month and 12 months per year, respectively. From this table, we can see that CMiS-010 is the top saver followed by EWoS-333 for the small-scale deployment scenario (i.e., using 40 gNBs and 8 VMs). This means that EWoS-333 achieves more than 95% of the ultimate maximum possible savings resulting from a pure cost minimization strategy. Using EWoS-333, for such a small scale simulated network, an MNO can save 4560$ per year. Recall that this cost is quantified using the GCP pricing data [13].

D. Number of Active gNBs and Active VMs

Fig. 9 shows the number of active gNBs per hour during the day, according to the traffic profiles of both residential and business areas. We can observe that LTMaS-001 has all the gNBs active since it tends to maximize low-loaded gNBs. This is similar to SSS where all the gNBs are kept active. These two strategies have the biggest number of active gNBs. PMiS-100 has the least number of active gNBs since it aims to reduce the power consumption. However, we note that this number increases hourly during the busy period as each additionally served gNB from low-loaded ones increases the power. As for CMiS-010, as long as involved VMs have the needed baseband processing capacity to handle associated gNBs, these latters are

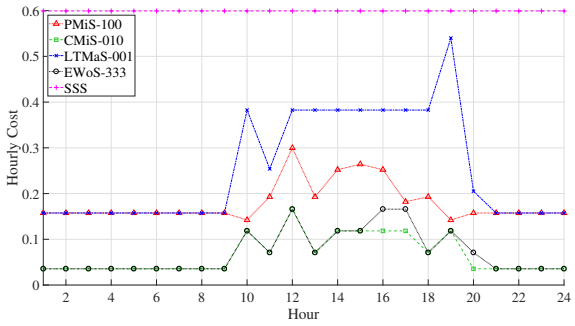

Fig. 7: CC costs ($) versus 24-hour time window

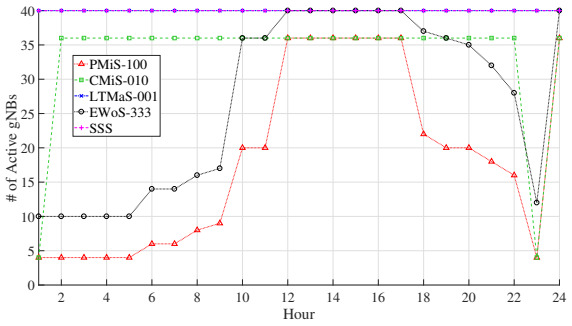

Fig. 9: Number of Active gNBs

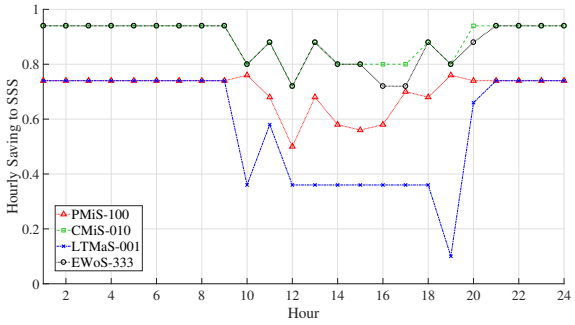

Fig. 8: Savings compared to SSS versus 24-hour time window

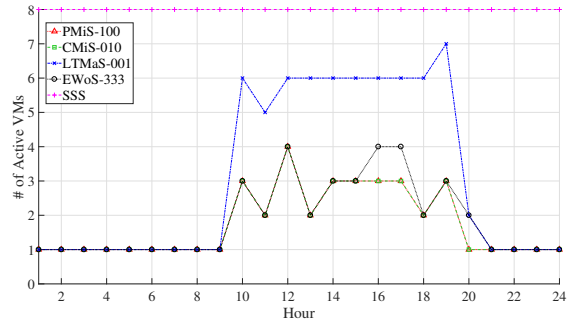

Fig. 10: Number of Active VMs versus 24-hour time window

kept active. This explains why its trend is higher than PMiS-100. EWoS-333, on the other hand, has less active gNBs than CMiS-010 before 10:00 and after 20:00 as it has to balance all the three different objectives.

Fig. 10 further investigates how VMs are activated during the day according to the studied traffic profiles. We can see that SSS keeps the totality of VMs active all the time which advocates the need of dynamicity for savings. LTMaS-001 is the second worst performing strategy as by maximizing the load of gNBs, the number of active VMs is also maximized. CMiS-010, EWoS-333 and PMiS-100 are closely performing. However, EWoS-333 tends to increase the number of active VMs at some hours, namely at 16:00 and 17:00 in order to meet throughput maximization objective.

E. Average Central Processing (CP) load per VM

In order to assess how the CP load is affected by dynamically mapping gNBs to VMs, we plot in Fig. 11 the average CP load per VM during the 24-hour time window. Interestingly, we can see that LTMaS-001 outperforms all the remaining schemes except for the period between 21:00 and 24:00, where the traffic resulting from business area is low. This is related to the fact that this approach aims to maximize the traffic load of low-loaded gNBs by increasing the number of VMs associated to those gNBs, which decreases the average CP load per VM. On one hand, PMiS-100 has the highest CP load per VM during the periods 8:00-10:00 and 14:00-20:00, since it aims to minimize the total processing power, which allows to minimize the number of instantiated VMs, increasing thus the average CP load per VM. However, the period from 11:00 to 14:00 is interesting; although business area traffic reaches its peak during this period, PMiS-100 has an average CP load lower than the three other strategies (i.e., CMiS-010, EWoS-333, and SSS). The reason is that the traffic of business area in this period is fluctuating in teeth saw pattern unlike the subsequent period from 14:00 to 17:00 where it has monotonous rising trend. On the other hand, SSS floats in between. CMiS-010 trails a little below PMiS-100 due to a higher number of served low-loaded gNBs whenever there is capacity in VMs.

F. Low-loaded gNBs silos and Average Number of VMs

Finally, Figs. 12 and 13 depict the percentage of low-loaded gNBs silos and average number of instantiated VMs according to each of our studied strategies. Recall first, that grouping the number of gNBs with low-load is an intuitive option to avoid investing power in a gNB rendering telecom services but having no users consuming them, which is a waste of resources. By grouping these gNBs, the service offering is not disrupted but the power consumption is decreased. In this simulation, we have varied the number of residential gNBs using a step of 10. As explained in the previous section, we can see that PMiS-100 has the least number of silo low-loaded gNBs, followed by EWoS-333 and then CMiS-010. The two strategies having biggest percentage of silo gNBs staying at 100% are LTMaS-001 and SSS as these two strategies aim to maximize the number of active gNBs at all time. Regarding

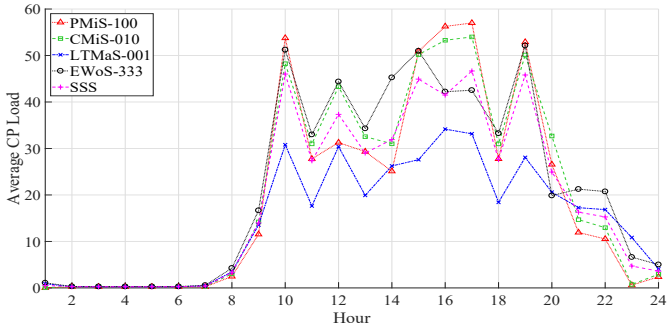

Fig. 11: Average CP Load per VM versus 24-hour time window

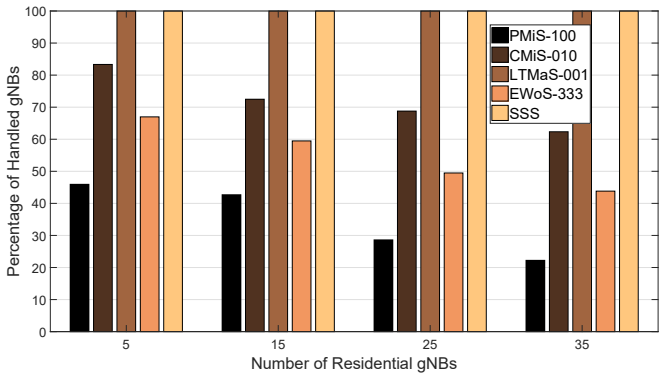

Fig. 12: Percentage of low-loaded gNBs silos

the average number of VMs to serve these gNBs, we can see, from Fig. 13 that the PMiS-100 and CMiS-010 are very close on this front followed by EWoS-333. LTMaS-001 and SSS are the worst two strategies as expected due to the maximized number of gNBs resulting from throughput maximization.

G. Final Remarks

To conclude our analysis of all the performance metrics, we report in Table IX the average relative saving of each scheme with respect to SSS. We can see that EWoS-333 provides a good trade-off as it is in between the PMIS-100 and CMIS-010 with 33% decrease of active gNBs. Interestingly, it is very close to PMIS-100 and CMIS-010 in terms of decrease of active VMs. This comes with a slight increase in CP load of 22.3% which is acceptable as long as it is less than the 100% maximum load of a VM. We can also observe that, by focusing only on the power minimization (i.e., PMIS-100), we can achieve high savings in term of number of active gNBs and active VMs, and a reasonable decrease in the CP load compared to the SSS scheme, with the expense of having high CP load compared to the other optimization strategies.

TABLE IX: Average saving (%) with respect to SSS

	PMIS-100	**CMIS-010**	**LTMaS-001**	**EWoS-333**
# active gNBs	54.06	16.67	0	33.02
# active VMs	78.13	78.13	60.94	76.56
CP Load	14.17	2.84	4.59	(-)22.3

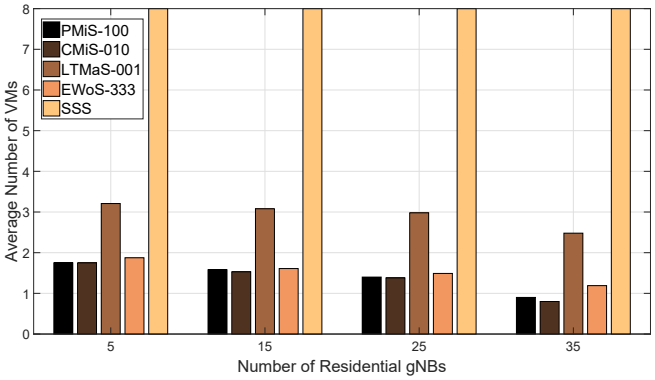

Fig. 13: Average Number of VMs versus number of residential gNBs

Based on the evaluation of all aforementioned performance metrics, we can conclude that an MNO can target multiple objectives at the same time and achieve more than 95% of a single cost minimization strategy by an adequate choice of parameters. We have seen that the EWoS-333 achieves thousands of $ yearly savings for a small-scale network. In addition, it performs very close to pure power minimization or cost minimization strategies in terms of decreasing the number of active VMs despite some acceptable increase of the CP load. These results provide guidelines that can be used by operators to decide the best optimization strategy according to their needs: processing power minimization, Cloud Computing cost minimization, network load maximization or all.

VI. Conclusion

In this paper, we have addressed the problem of mapping gNBs to VMs for Software-Defined Data Centers in 5G. We have formulated the problem as an ILP with a generic weighted objective function composed of three homogenized terms: processing power minimization, virtualization cost minimization, and low-loaded gNBs traffic load maximization. We then proposed a framework to solve this problem using Branch, Cut and Price (BCP) algorithm. We evaluated and analyzed different strategies in terms of each of the targeted objectives. Such strategies have shown different facets of their pros/cons so that MNOs could select the best strategy suiting their needs. Surprisingly, we found out that the EWoS-333 strategy, which gives equal weights for the three objectives, performs very close to the pure cost minimization approach and outperforms other strategies thanks to the well-balanced competing objectives. Particularly, in our setup, we found out that EWoS-333 achieves more than 95% of the ultimate maximum possible savings resulting from a pure cost minimization strategy at a price of acceptable slight Central Processing (CP) load increase. Also, results show that proposed BCP-based MPDC performs extremely well in term of computation time compared to naive exhaustive search and better than alternative strategies using Branch-and-Bound and Branch-and-Cut.

ACKNOWLEDGMENT

This work was supported by the FUI SCORPION project (Grant no. 17/00464), CNRS PRESS (Grant no. 07771), "Azm & Saade Foundation", and Lebanese University.

REFERENCES

[1] "3GPP TS 32.130: Network Sharing: Concepts (Rel. 14)," Dec. 2016.
[2] CSA and McAfee, "WP custom apps IaaS trends," Tech. Rep., 2017.
[3] Gartner, "Forecasts worldwide public cloud revenue to grow," 2-April-2019. [Online]. Available: https://gartner.com
[4] Microsoft, "AT&T and Microsoft announce a strategic alliance to deliver innovation with cloud, AI and 5G," Jul 2019.
[5] Ovum, "Understanding the Business Value of Re-architecting Core Applications on the Public Cloud," Feb 2019.
[6] McKinsey, "Creating value with the cloud," December-2018.
[7] "3GPP TS 22.261: "Service requirements for the 5g system ," Jun. 2017.
[8] N. Alliance, "Optimised backhaul requirements," *Next Generation Mobile Networks Alliance*, p. 19, 2008.
[9] "3GPP TR 29.890, Technical Specification Group Core Network and Terminals; Study on CT WG3 Aspects of 5G System Ph.1; R.15," 2018.
[10] "3GPP TS 23.501 System Architecture for the 5G System; St.2," 2018.
[11] NGMN, "Service-Based Architecture in 5G," January 2018.
[12] "ETSI GR NFV-IFA 029 V0.8.0, Report on the Enhancements of the NFV architecture towards Cloud-Native and PaaS," 2018.
[13] Google, "Google cloud pricing list," 2018, [accessed 5-September-2018]. [Online]. Available: https://cloud.google.com/compute/pricing
[14] M. Bouet, J. Leguay, T. Combe, and V. Conan, "Cost-based placement of vDPI functions in nfv infrastructures," *International Journal of Network Management*, vol. 25, no. 6, pp. 490–506, 2015.
[15] G. Liu and H. Shen, "Minimum-cost cloud storage service across cloud," *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, 2017.
[16] Z. Wu, M. Butkiewicz, D. Perkins, E. Katz-Bassett, and H. V. Madhyastha, "Spanstore: Cost-effective geo-replicated storage spanning multiple cloud services," in *Proceedings of the 24th ACM Symposium on Operating Systems Principles*, 2013.
[17] Y. Ran, J. Yang, S. Zhang, and H. Xi, "Dynamic IaaS computing resource provisioning strategy with QoS constraint," *IEEE Transactions on Services Computing*, vol. 10, no. 2, 2017.
[18] Q. Wang, M. M. Tan, X. Tang, and W. Cai, "Minimizing cost in IaaS clouds via scheduled instance reservation," in *Distributed Computing Systems (ICDCS), IEEE 37th International Conference*, 2017.
[19] M. Qian, W. Hardjawana, J. Shi, and B. Vucetic, "Baseband processing units virtualization for cloud radio access networks," *IEEE Wireless Communications Letters*, vol. 4, no. 2, pp. 189–192, 2015.
[20] T. Kuo, B. Liou, K. C. Lin, and M. Tsai, "Deploying chains of virtual network functions: On the relation between link and server usage," in *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, April 2016.
[21] E. Baccarelli, M. Scarpiniti, and A. Momenzadeh, "Design and dynamic optimization of a 5g mobile-fog-cloud multi-tier ecosystem for the real-time distributed execution of stream apps," *IEEE Access*, vol. 7, 2019.
[22] C. Dong, W. Wen, T. Xu, and X. Yang, "Joint optimization of data-center selection and video-streaming distribution for crowdsourced live streaming in a geo-distributed cloud platform," *IEEE Transactions on Network and Service Management*, vol. 16, no. 2, June 2019.
[23] T. Wang, F. Liu, and H. Xu, "An efficient online algorithm for dynamic sdn controller assignment in data center networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, Oct 2017.
[24] Y. T. Woldeyohannes, A. Mohammadkhan, K. K. Ramakrishnan, and Y. Jiang, "Cluspr: Balancing multiple objectives at scale for NFV," *IEEE Transactions on Network and Service Management*, vol. 15, Dec 2018.
[25] A. Basta and et al., "Towards a cost optimal design for a 5G mobile core network," *IEEE Transactions on Network and Service Management*, vol. 14, no. 4, Dec 2017.
[26] S. Mireslami, L. Rakai, B. H. Far, and M. Wang, "Simultaneous cost and qos optimization for cloud resource allocation," *IEEE Transactions on Network and Service Management*, vol. 14, no. 3, Sep. 2017.
[27] Amazon Web Services, "Aws - it service management company." [Online]. Available: https://aws.amazon.com/
[28] Rackspace, "Managed dedicated and cloud computing services." [Online]. Available: https://www.rackspace.com/
[29] B. Kar, E. H. Wu, and Y. Lin, "Energy cost optimization in dynamic placement of virtualized network function chains," *IEEE Transactions on Network and Service Management*, vol. 15, no. 1, March 2018.
[30] M. Mrad, A. Balma, F. Moalla, and T. Ladhari, "Nodes migration scheduling of access networks," *IEEE Transactions on Network and Service Management*, vol. 14, no. 1, March 2017.
[31] P. Avella, M. Boccia, and I. Vasilyev, "A branch-and-cut algorithm for the multilevel generalized assignment problem," *IEEE Access*, 2013.
[32] M. Y. Lyazidi, L. Giupponi, J. Mangues-Bafalluy, N. Aitsaadi, and R. Langar, "A novel optimization framework for C-RAN BBU selection," in *86th IEEE Vehicular Technology Conference (VTC-Fall)*, 2017.
[33] G. Gamrath, "Generic branch-cut-and-price," 2010.
[34] H. A. Alameddine, S. Sebbah, and C. Assi, "On the interplay between network function mapping and scheduling in VNF," *IEEE Transactions on Network and Service Management*, vol. 14, no. 4, Dec 2017.
[35] E. Pateromichelakis and et al., "Service-tailored user-plane design framework and architecture considerations in 5G," *IEEE Access*, vol. 5, 2017.
[36] H. Wang, F. Xu, Y. Li, P. Zhang, and D. Jin, "Understanding mobile traffic patterns of large scale cellular towers in urban environment," in *Proceedings of the 2015 Internet Measurement Conference*. ACM, 2015.
[37] L. Nirenberg, "The weyl and minkowski problems in differential geometry," *Communications on pure and applied mathematics*, vol. 6, 1953.
[38] C. Barnhart, C. A. Hane, and P. H. Vance, "Using branch-and-price-and-cut to solve origin-destination integer multicommodity flow problems," *Operations Research*, vol. 48, no. 2, pp. 318–326, 2000.
[39] China Mobile Research Institute, "C-RAN, The Road Towards Green RAN, White Paper,Version 3.0," Apr 2013.
[40] "IBM CPLEX optimizer studio ide version 12.8," https://www.ibm.com/analytics/cplex-optimizer.
[41] A. Jindal, V. Podolskiy, and M. Gerndt, "Performance modeling for cloud microservice applications," in *Proceedings of the 2019 ACM/SPEC International Conference on Performance Engineering*. ACM, 2019.
[42] F. Quesnel, H. K. Mehta, and J. Menaud, "Estimating the power consumption of an idle virtual machine," in *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*.
[43] VMWARE Docs, "About Allocating Bandwidth for Virtual Machines," Apr 2014. [Online]. Available: https://docs.vmware.com/

Nazih Salhab received his Computer science and Telecommunication Engineer degree from the Lebanese University in 2003. He has more than 15 years of experience in mobile networks operation management, business analysis, and project management for major telecom network operators. He was invited to several international labs as visiting scientist, including EPFL-Switzerland and D.R. Cheriton, University of Waterloo, ON, Canada. He is pursuing a double PhD degree in University Gustave Eiffel (UGE) and the doctoral school of science and technology of the Lebanese University.

Rana Rahim received her Computer science and Telecommunication Engineer degree from the Lebanese University in 2002. She then obtained her Master degree (DEA) in 2003 from the USJ University (Lebanon) and the Lebanese University, and her PhD degree in January 2008 from the University of Technology of Troyes (UTT) - France. She was a postdoctoral researcher at the UTT from October 2008 to October 2009. She obtained her HDR (Habilitation à Diriger des Recherches) in 2016. She is currently Associate professor at the Lebanese University. Her research interests include System management, Quality of Service, IoT Networks, Smart Grids, cloud radio access networks, slicing in 5G and software-defined networks.

Rami Langar is currently a Full Professor at University Gustave Eiffel (UGE), France. Before joining UGE, he was an Associate Professor at LIP6, University Pierre and Marie Curie (UPMC, now Sorbonne Universite) between 2008 and 2016, and a Post-Doctoral Research Fellow at the School of Computer Science, University of Waterloo, ON, Canada between 2006 and 2008. He received the M.Sc. degree in network and computer science from UPMC in 2002; and the Ph.D degree in network and computer science from Telecom ParisTech, Paris, France, in 2006. He was chair of IEEE ComSoc Technical Committee on Information Infrastructure and Networking (TCIIN) for the term Jan. 2018-Dec. 2019. His research interests include resource management in future wireless systems, Cloud-RAN, network slicing in 5G/5G+, software-defined wireless networks, smart cities, and mobile Cloud offloading.