Offloading Network Data Analytics Function to the Cloud with Minimum Cost and Maximum Utilization

Nazih Salhab*¶, Rana Rahim‡, Rami Langar*, Raouf Boutaba§
* University Gustave Eiffel, LIGM-CNRS UMR 8049, F-77454, Marne-la-Vallée, France
‡ LTRM, Faculty of Science, Lebanese University, Tripoli, Lebanon
¶ LTRM, Doctoral School of Science and Technology, Lebanese University, Lebanon
§ David R. Cheriton School, University of Waterloo, Waterloo, ON, Canada
E-mails: nazih.salhab@univ-eiffel.fr; rana.rahim@ul.edu.lb; rami.langar@univ-eiffel.fr, rboutaba@uwaterloo.ca

***Abstract*—Cloud computing is being embraced more and more by telecommunication operators for on-demand access to computing resources. Knowing that 5G Core reference architecture is envisioned to be cloud-native and service-oriented, we propose, in this paper, offloading to the cloud, some of 5G delay-tolerant Network Functions and in particular the Network Data Analytics Function (NWDAF). The dynamic selection of cloud resources to serve off-loaded 5G-NWDAF, while incurring minimum cost and maximizing utilization of served next generation Node-Bs (gNBs) requires agility and automation. This paper introduces a framework to automate the selection process that satisfies resource demands while meeting two objectives, namely, cost minimization and utilization maximization. We first formulate the mapping of gNBs to 5G-NWDAF problem as an Integer Linear Program (ILP). Then, we propose an algorithm to solve it based on branch-cut-and-price technique combining all of branch-and-price, branch-and-cut and branch-and-bound. Results using pricing data from a public cloud provider (Google Cloud Platform), show that our proposal achieves important savings in cloud computing costs and reduction in execution time compared to other state-of-the-art frameworks.**

***Index Terms*—Cloud Computing, 5G Core Network Offloading, Branch-Cut-and-Price, Multi-objective optimization, Google Cloud Platform**

I. Introduction

Cloud Computing (CC) is getting popularity among Telephone Companies (telcos). AT&T stated that it is becoming a ‘public cloud first’ company by migrating its workloads to Microsoft public cloud by 2024. They advocate that this allows them to focus on core network capabilities, accelerate their innovation cycle, and empower their workforce while optimizing costs [1]. TM-Forum claims that telcos cannot afford not to embrace the public clouds [2]. Surveys show that enterprises are divesting their data centers and moving application workloads, both testing and production to the public cloud [3]. As of January 2017, 46.1% of business-critical applications are in the public or hybrid cloud [3]. Gartner forecasts cloud services industry to grow exponentially through 2022 [4]. Furthermore, a leading research and consulting business mandates that in order to compete in the digital world, the adoption of public cloud by telcos is inevitable [5]. Readings show that telcos are among the fastest-growing users of public cloud computing starting from 2020 as they look to accelerate their new service delivery plan [5]. Indeed, usage of CC allows telcos to move faster, focus on their core business, minimize their hardware footprints, and keep pace with increasing demands of resources. This is due to inherent cloud elasticity and versatility to provide resources as needed. In addition, by leveraging auto-scaling capabilities of the public cloud, telcos will pay only for what they need when they need it. With the competition from Over-The-Top providers, telcos have to minimize their costs to maintain profitability [6]. To afford the tremendous communications infrastructure overhaul that 5G requires, telcos need to create additional revenue generating services such as data analytics. One way to achieve this goal is by exploiting the cloud to deploy remote Network Functions (NFs) for delay-tolerant services [6]. Not only NFs offloading to the cloud is a way to cut costs, but also, it serves as a driver for new business models for telcos, especially on data analytics front. Network Data Analytics Function (NWDAF) [7], part of 5G Core, is supposed to crunch huge amounts of data and report analytics outcomes to multiple NFs. As of today, the Network Slice Selection Function (NSSF) and Policy Control Function (PCF) are consumers of NWDAF, but according to 3GPP standard, any NF or NF-service can consume it too [8]. In this paper, we use “Compute” resources in public clouds, expressed in Virtual Central Processing Units (vCPUs) and Virtual Memory (vMEM) to implement 5G-NFs and in particular, edge NWDAFs.

Our objective is to dynamically deploy Virtual Machines (VMs) on the cloud to implement 5G-NF at minimum cost with maximum utilization according to the load of the new generation Node-Bs (gNBs) as depicted in Fig. 1.

Our contributions are summarized as follows:

- We model the selection of 5G-NF VMs, while minimizing CC cost and maximizing utilization to serve gNBs and formulate it as an Integer Linear Program (ILP).
- We propose an algorithm using several techniques, namely branch-and-bound, branch-and-price and branch-and-cut to solve our ILP problem.
- We show the effectiveness of our proposal compared to

978-1-7281-5089-5/20/$31.00 ©2020 IEEE

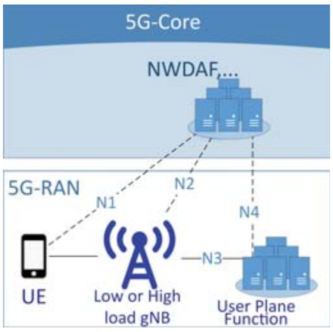


Fig. 1: Simplified gNB-NWDAF Architecture

other solutions using pricing data from Google Cloud Platform (GCP).

The remainder of this paper is organized as follows. In section II, we discuss the related works. Section III describes the system model and formulates our problem as an ILP. Our proposed algorithm to solve the ILP problem is presented in section IV. Section V provides performance evaluation including assumptions validation and simulation results discussion. We conclude this paper in Section VI.

II. Related Works

Minimizing cost when using CC has triggered considerable interest among researchers.

Authors in [9] proposed cost minimization using storage across multiple cloud providers, while meeting multiple service level objectives. Also, authors in [10] proposed to minimize cloud storage costs, while achieving latency and availability objectives across multiple Cloud Service Providers (CSPs). Both of these papers treated the cost optimization from "Storage" resources minimization perspective. Using storage cannot be applied to stateless applications, where no data need to be stored. Differently from them, we focus on compute resources minimization.

Authors in [11] proposed a dynamic approach to predict the load using autoregressive model to calculate the number of instances to be reserved for average computation requirements. Authors in [12] proposed a CC cost saving by exploiting the discounts resulting from scheduling reservation of resources on recurring basis in advance. Unlike these two approaches, we do not rely on prediction to save costs but on dynamically optimizing cloud resource selection over time.

In [13], authors proposed dynamic placement of virtual deep packet inspection function in NFV infrastructure to minimize operational expenditures including licensing cost and power consumption. They formulated this problem as multi-commodity flow Integer Linear Programming (ILP) and proposed a centrality-based greedy heuristic that runs in polynomial time. Unlike this work, we consider, in our paper, the utilization in addition to the cost to meet telcos optimization strategy.

Authors in [14] proposed a Branch and Bound (BB) approach for resource constrained scheduling in two phases to reduce the computation time.

Authors in [15] proposed a multilevel generalized assignment problem for minimizing the assignment cost of jobs to machines using Branch and Cut (BC). Authors in [16] formulated Cloud Radio Access Network Assignment problem as an ILP and used Branch and Price (BP) to solve and evaluate different strategies for a multi-objective optimization. Different from these works, we focus, in this paper, on minimizing CC costs and maximizing utilization of gNBs and propose an efficient algorithm to solve the 5G-NF selection problem using a Branch, Cut and Price (BCP) approach.

III. System model and problem formulation

A. System Model

We consider a simplified 5G architecture consisting of three domains, including: Radio Access Network (RAN), 5G-NF and a backhaul transport network for interconnecting the RAN to the 5G-NF. A 5G-NF Service overlaying a number of N gNBs needs to be deployed on a pool of M VMs. The pool of VMs is denoted by $\mathbb{V} = \{i|1 \leq i \leq M\}$. We assume that the latency imposed by hosting the 5G-NF for delay tolerant services on the cloud is acceptable when backhauling gNBs to the cloud. Indeed, according to the requirement R48 of the NGMN alliance [17], an end-to-end latency of 10 milliseconds is considered fine for most critical applications such as voice and video over IP. This is easily achievable nowadays in most public clouds as we will confirm in the performance evaluation section (cf. section V). The set of gNBs is denoted by $\mathbb{G} = \{j|1 \leq j \leq N\}$. A group of gNBs is associated to one 5G-NF VM pool. We define a binary decision variable denoted by r_{ij} to show if $\mathrm{VM_i}$ is associated to $\mathrm{gNB_j}$ or not. The average utilization y_i of the VM pool i is formulated as follows.

$$y_i = \frac{1}{C_i}\sum_{j=1}^{N} r_{ij}.l_j \tag{1}$$

where l_j denotes the traffic utilization in vCPUs on the $\mathrm{gNB_j}$ and C_i denotes the maximum capacity in vCPUs of the $\mathrm{VM_i}$ implementing a 5G-NF. We model the price of instantiating VMs to implement the 5G-NF pool i as a function of the average utilization of VMs y_i expressed in (1). We use a linear model [18] with a proportionality slope (λ) as:

$$P_i = \lambda.y_i + P_0. \tag{2}$$

P_0 is the fixed price portion imposed by the CSP. To normalize the price elaborated in (2), we denote by P_{max} the highest value of the VM in the CSP pricing list. We consider two sets of gNBs, formed according to the traffic load of each gNB. They are $\mathbb{G_L}$ for low-load gNBs and $\mathbb{G_H}$ for high-load gNBs. We propose this segregation of gNBs based on traffic load, because we anticipate to have asymmetrical traffic between day and night in addition to differences between business and residential areas in term of processing capacity requirement for services [16]. We finally define a binary mapping variable x_i to express the active state of a $\mathrm{VM_i}$ such that x_i = 0 when

$\sum_{j=1}^{N} r_{ij} = 0$, meaning that no $\mathrm{gNB_j}$ is mapped to a $\mathrm{VM_i}$ for all i, j and $x_i = 1$, otherwise.

B. Problem Formulation

We define two parameters α and β as weight coefficients with values ranging between 0 and 1 so that we scalarize our multi-objective Minimum Cost, Maximum Utilization (MCMU) problem. We assume that these parameters are set by the operator to specify the sought optimization strategy according to the choice of prevailing factors (CC cost or utilization maximization from low-loaded gNBs). We formulate our MCMU problem as a weighted optimization problem with two homogenized objective terms as follows.

$$\min_{r} \quad \alpha \sum_{i=1}^{M} \frac{x_i P_0 + \lambda y_i}{P_{max}} - \beta \sum_{i=1}^{M} \frac{\sum_{j \in \mathbb{G}_{\mathbb{L}}} r_{ij} l_j}{\sum_{j \in \mathbb{G}_{\mathbb{L}}} l_j} \tag{3a}$$

s.t.

$$\sum_{i=1}^{M} \sum_{j \in \mathbb{G}_{\mathbb{H}}} r_{ij} l_j = \sum_{j \in \mathbb{G}_{\mathbb{H}}} l_j \tag{3b}$$

$$\sum_{j=1}^{N} r_{ij} l_j \leq C_i, \forall i \in \{1, \ldots, M\} \tag{3c}$$

$$\sum_{i=1}^{M} r_{ij} \leq 1, \forall j \in \{1, \ldots, N\} \tag{3d}$$

$$x_i \in \{0,1\}, \forall i \in \{1, \ldots, M\} \tag{3e}$$

$$r_{ij} \in \{0,1\}, \forall i \in \{1, \ldots, M\}, \forall j \in \{1, \ldots, N\} \tag{3f}$$

The proposed objective function in (3a) consists of minimizing the total VM pool operation cost and maximizing the traffic utilization resulting from the low load traffic gNBs, while entirely satisfying the high-load traffic gNBs. Indeed, constraint (3b) specifies that the traffic of highly loaded gNBs is totally handled by the VMs implementing the 5G-NF. Constraint (3c) ensures that the capacity (C_i) of the VM pool i is not exceeded by the sum of load of its children gNBs. Constraint (3d) stipulates that no gNB could be associated to more than one VM pool of 5G-NF. Constraints (3e) and (3f) stipulate that the decision variables are binary.

IV. Proposed Algorithm

Our MCMU problem, formulated in (3), is an ILP and hence cannot be solved directly using convex optimization techniques. It is NP-hard and the optimal solution can only be found by exhaustively figuring out all M^N possible combinations of VM/gNB assignments which is impractical for large-scale networks [16]. Therefore, we propose an algorithm based on the BCP framework [19], by combining column generation starting from linear relaxation, along with using cut planes before resorting to branch-and-bound to compute the optimal solution of our MCMU problem. Linear relaxation is about disregarding the integrality constraint of integer variables. Cuts attempt to restrict the feasible region of the linear relaxations so that their solutions are closer to integers. In the BCP algorithm, sets of columns are left out of the linear relaxation in order to handle the problem more efficiently by decreasing the computational complexity. Columns are then "priced" and added back to the linear relaxation as needed. To decide which column will be added, a sub-problem called the "pricing problem" is created to identify which columns should enter the basis in an aim to decrease the objective function in case of minimization. When such column is found, the Linear Program (LP) is then re-optimized.

In the following subsection, we detail the steps of our BCP algorithm first by formalizing the steps for Column Generation on our MCMU problem by means of a problem transformation, and then decomposing it into Master (MP) and Pricing (PP) problems as follows.

A. Problem Transformation

Based on the structure of our original problem and using Minkowski-Weyl's representation theorem [20] stating that every polyhedron $\mathbb{P}$ can be represented in the form of a convex linear expression of extreme points v and extreme rays w, we transform our original problem as follows. Recall first that this theorem states that $\mathbb{P} = \{\forall r \in \mathbb{R}^n, \exists (\rho, \mu) \in \mathbb{R}^2 : r = \sum \rho.v + \sum \mu.w\}$ where ρ, μ are linear coefficients. Instead of the initial decision variable r_{ij}, we use two binary variables v_{ij} and w_{ij}, for the gNBs with low (denoted as l_j^L) and high traffic-load (denoted as l_j^H), respectively. Same definition remains for x_i after this transformation, i.e., $x_i = 0$ if $\mathrm{VM_i}$ is inactive ($\sum_{j \in \mathbb{G}_{\mathbb{L}}} v_{ij} + \sum_{j \in \mathbb{G}_{\mathbb{H}}} w_{ij} = 0$) and 1 otherwise. This way, our MCMU problem becomes as follows.

$$\min_{v,\, w} \quad \Phi \sum_{i=1}^{M} x_i + \sum_{i=1}^{M} \sum_{j \in \mathbb{G}_{\mathbb{L}}} \Omega_i v_{ij} l_j^L + \sum_{i=1}^{M} \sum_{j \in \mathbb{G}_{\mathbb{H}}} \Psi_i w_{ij} l_j^H \tag{4a}$$

s.t.

$$\sum_{i=1}^{M} \sum_{j \in \mathbb{G}_{\mathbb{H}}} w_{ij} l_j^H = \sum_{j \in \mathbb{G}_{\mathbb{H}}} l_j^H \tag{4b}$$

$$\sum_{j \in \mathbb{G}_{\mathbb{L}}} v_{ij}.l_j^L + \sum_{j \in \mathbb{G}_{\mathbb{H}}} w_{ij}.l_j^H \leq C_i, \forall i \in \{1, \ldots, M\} \tag{4c}$$

$$\sum_{i=1}^{M} v_{ij} \leq 1, \forall j \in \mathbb{G}_{\mathbb{L}} \tag{4d}$$

$$\sum_{i=1}^{M} w_{ij} \leq 1, \forall j \in \mathbb{G}_{\mathbb{H}} \tag{4e}$$

$$v_{ij} \in \{0,1\}, \forall i \in \{1, \ldots, M\}, \forall j \in \mathbb{G}_{\mathbb{L}} \tag{4f}$$

$$w_{ij} \in \{0,1\}, \forall i \in \{1, \ldots, M\}, \forall j \in \mathbb{G}_{\mathbb{H}} \tag{4g}$$

where $\Phi = \frac{\alpha P_0}{P_{max}}$, $\Omega_i = \frac{\alpha \lambda}{C_i.P_{max}} - \frac{\beta}{\sum_{j \in \mathbb{S}_{\mathbb{L}}} l_j^L}$ and $\Psi_i = \frac{\alpha \lambda}{C_i.P_{max}}$. Let the two sets of feasible possible assignments of low and high-traffic load gNBs to VM pool i be $\Xi_i^L = \{v_1^i, v_2^i, \ldots, v_{k_i}^i\}$ and $\Xi_i^H = \{w_1^i, w_2^i, \ldots, w_{k_i}^i\}$. We suppose that two particular variables of Ξ_i^L and Ξ_i^H, $v_k^i = \{v_{1k}^i, v_{2k}^i, \ldots, v_{Sk}^i\}$ and $w_k^i = \{w_{1k}^i, w_{2k}^i, \ldots, w_{Sk}^i\}$ are a valid solution to our transformed problem formulated in (4). Based on Dantzig-Wolfe's decomposition [21] that 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-dimensional decision variable, that reflects the feasibility of the selected solution. Accordingly, z_k^i would be equal to (1,1) when (v_k^i, w_k^i) is feasible and (0,0) othwerwise. The Master Problem (MP) is a sub-version of the transformed problem, where we disregard the complicating (coupling) constraints (4c). MP is then expressed as follows.

$$\text{(MP)} \min_{z} \quad \sum_{k=1}^{k_i}\sum_{i=1}^{M}(\Phi x_i + \sum_{j\in\mathbb{G}_\mathbb{L}} \Omega_i v_{ij} l_j^L \dot{z}_k^i + \sum_{j\in\mathbb{G}_\mathbb{H}} \Psi_i w_{ij} l_j^H \ddot{z}_k^i) \tag{5a}$$

s.t.

$$\sum_{k=1}^{k_i}\sum_{i=1}^{M}\sum_{j\in\mathbb{G}_\mathbb{H}} \ddot{z}_k^i w_{jk}^i l_j^H = \sum_{j\in\mathbb{G}_\mathbb{H}} l_j^H \tag{5b}$$

$$\sum_{k=1}^{k_i} \dot{z}_k^i \leq 1, \sum_{k=1}^{k_i} \ddot{z}_k^i \leq 1, \forall i \in \{1,\dots,M\} \tag{5c}$$

$$\sum_{k=1}^{k_i}\sum_{i=1}^{M} \dot{z}_k^i v_{ij} \leq 1, \forall j \in \mathbb{G}_\mathbb{L} \tag{5d}$$

$$\sum_{k=1}^{k_i}\sum_{i=1}^{M} \ddot{z}_k^i w_{ij} \leq 1, \forall j \in \mathbb{G}_\mathbb{H} \tag{5e}$$

$$\dot{z}_k^i, \ddot{z}_k^i \in \{0,1\}, \forall i \in \{1,\dots,M\}, k \in \{1,\dots,k_i\} \tag{5f}$$

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

$$\min_{1\leq i \leq M}\{o^i - z^{*i}\} \tag{6}$$

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

$$\text{(PP)} \min_{v,w} \quad \Phi x_i + \sum_{j\in\mathbb{G}_\mathbb{L}} \Omega_i v_j^i (l_j^L - v_j^*) + \sum_{j\in\mathbb{G}_\mathbb{H}} \Psi_i w_j^i (l_j^H - w_j^*) \tag{7a}$$

s.t.

$$\sum_{j\in\mathbb{G}_\mathbb{L}} v_j^i l_j^L + \sum_{j\in\mathbb{G}_\mathbb{H}} w_j^i l_j^H \leq C_i, \forall i \in \{1,\dots,M\} \tag{7b}$$

$$v_{ij}, w_{ij} \in \{0,1\}, \forall i \in \{1,\dots,M\}, \forall j \in \{1,\dots,N\} \tag{7c}$$

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 traffic load gNBs. In the PP, we get the optimum mapping of gNBs to VM pool i.

B. Proposed MCMU algorithm

Algorithm 1: BCP-based MCMU

Data: Objective function and constraints
Result: gNBs to VM pool mapping solution
Initialize our problem
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 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 criteria is reached* **then** quit;
goto (A)

To find a solution to our original problem, we propose an algorithm that achieves optimal results with a noticeable gain in computation time, especially for large problem instances depending on the values of M and N. Our algorithm is listed in Algorithm 1. We start by generating an initial set of configurations. Next, we apply LP relaxation to our problem (P) and solve the LP. We iterate to complement found columns to the basis of our solution. Then, we proceed to cut generation, and we try to find integer-feasible solutions before we use branch-and-bound to systematically search for the optimal solution as long as the stop criterion is not reached. Stop criterion could be either a time-limit or a relative gap tolerance between the found value and LP value.

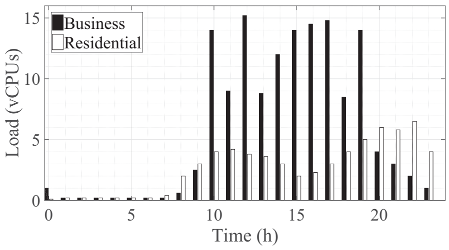


Fig. 2: gNBs Traffic Load versus Time

V. Performance Evaluation

In this section, we evaluate the performance of our MCMU algorithm using CPLEX Optimizer [22]. Note that, according to the values of α and β used in the objective function formulated in (3a), we refer to our algorithm as $MCMU\alpha\beta$ omitting decimal points from α and β.

Simulation parameters are reported in Table I. On the RAN Side, we consider a total of N = 2000 gNBs, including 1500 business and 500 residential gNBs, adapted from an hourly traffic load from [16] and [23] by considering a linear relation between the load in Mbps and the number of needed vCPUs as shown in Fig. 2. We conducted 200 experiments with changing VM pool size in vCPUs from 200 vCPUs to 2200 with a step of 10 vCPUs. For ease of interpretation, we split these 200 experiments into three categories and we average each category and denote it by: small, medium and large setup, respectively. GCP offers different machine types: standard (std), micro and small in addition to types that are highly performing in terms of vCPUs or vMEM. For each type, different discrete options exist in term of count of vCPUs (1..96). Accordingly, hourly prices are charged as functions of chosen specifications.

A. Data preparation and assumption validation

Pricing is fetched from Google Compute Engine (GCE) [24] that is the "compute" service from GCP. Same data is found for other cloud providers such as Amazon AWS or Microsoft Azure. We have chosen to conduct our simulations using GCE pricing data because GCP offers low latency within the stipulated limit of NGMN on the backhaul. To validate this assumption, we instantiated the smallest VM instance, called (f1-Micro), using Ubuntu 16.04 on major European regions covered by GCP and generated within each VM 100 ping messages to other public IPs of instantiated VMs. After averaging, we found that the ping takes less than 10 ms between several points of presence in Europe, as reported in Table II.

A default setup is proposed by GCE [25] for VMs. It has a Standard Price (SP) which we use as our baseline. We also computed the lower-bound solution of VM minimizing the cost and maximizing the utilization by solving the LP problem.

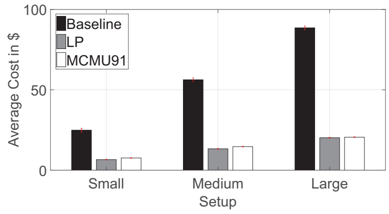


Fig. 3: Cost Comparison of Baseline, LP, MCMU91

TABLE I: Simulation parameters

Parameter	Value
Machine type	std, highcpu, highmem, megamem, ultramem, micro and small
VM vCPUs standard sizes	1, 2, 4, 8, 16, 32, 40, 64, 80, 96
Memory (GB)	0.6 .. 3844
Hourly Price (USD)	$0.0076 \ldots 27.7557$
(α, β)	(0.9, 0.1), (0.5, 0.5), (0.1, 0.9)
small/medium/large setup	525/1185/1855 (vCPUs)
Count of gNBs (N)	2000
Business gNBs	1500
Residential gNBs	500
Max Capacity C per VM	96 (vCPUs) [24]
Number of Experiments	200
stop criterion	120 seconds

B. Results

Fig. 3 depicts the categorized hourly average costs for all schemes (Baseline, LP, and MCMU91). We can see that MCMU91 dramatically decreases the average cost compared to the baseline (SP) and provides values that are close to the LP. The reason is that the baseline scheme selects the standard VM by default without considering the gNBs load and the matching VM capacity.

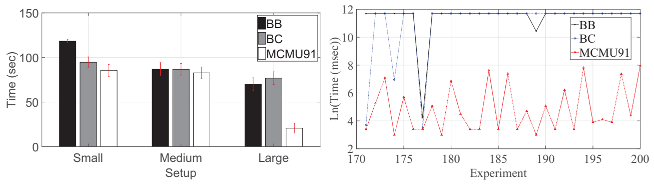
In order to assess the effectiveness of our algorithm, we considered 200 experiments with variable C_i from 200 to 2190 with a step size of 10 and we classified these experiments into 3 setups (small, medium and large) according to the values of C_i. Accordingly, each setup consists of 66 experiments. Fig. 4a compares the average computation time of MCMU with two well-known algorithms: BB [14], and BC [15], for the three setups of VM pool sizes. We can see that our MCMU algorithm is faster than BB and BC approaches, especially for large setups. For the small setups, as the constraints are aggressive in term of gNBs to VM pool mapping, we found that for some cases, the three evaluated approaches (BB, BC, and MCMU91) were not able to find a solution in a timely manner and thus the stop criteria of 120 seconds is reached which explains the increase in time. For medium setups, the ability to find a solution for all the approaches is comparable. Note that BC performs worst than BB in the large scenario

TABLE II: Latency in ms for some countries using 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

case and it could not find a solution before the stop criterion for several experiments.

To further show the effectiveness of our proposal, we plot in Fig. 4b the time taken by our algorithm compared to BB and BC in the last 30 experiments of large setups. We measured the time in milliseconds and plotted them in logarithmic scale as there is one order of magnitude difference. We see that for the majority of the experiments, BB and BC could not find a solution before the chosen stop criterion, while MCMU based on BCP could find it, thanks to its faster convergence resulting from combining column generation and cuts on top of BB.



(a) Multiple methods (b) Zoom on the computation time

Fig. 4: Computation Time Comparison

Fig. 5 shows the impact of the two parameters α and β on the performance of our MCMU algorithm. We considered different optimization strategies according to the chosen values of these two parameters as summarized in table III. We can see

TABLE III: Optimization Strategies

Scheme	**Values of** α, β
Prevailing Cost over Utilization (MCMU91)	0.9, 0.1
Equal-Importance of Cost and Utilization (MCMU55)	0.5, 0.5
Prevailing Utilization Maximization (MCMU19)	0.1, 0.9

that MCMU91 performs the best as the importance is given to the cost. MCMU55 gives equal importance to each of the weight factors and consequently lags behind MCMU91 and comes ahead of MCMU19 where the cost is maximum. The reason is that MCMU55, although it provides proportional fairness in regard to each of the objectives but it increases the incurred cost.

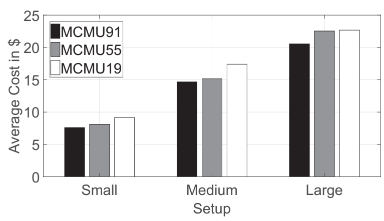


Fig. 5: MCMU Schemes with different α and β

VI. Conclusion

This paper addressed the minimum cost maximum utilization optimization problem for offloading delay tolerant 5G Network Functions (e.g. NWDAF) to public clouds. We formulated this problem as an Integer Linear Program and proposed a simple yet efficient algorithm based on the branch-cut-and-price framework to solve it. Results show that our algorithm performs well compared to optimal solution by providing considerable cost saving compared to the standard problem of VM provisioning with default VM selected. Also, using simulations, we found that our algorithm is faster and more likely to find a solution compared to other state-of-the-art approaches.

Acknowledgement

This work was supported by FUI SCORPION project (Grant no. 17/00464), CNRS PRESS project (Grant no. 239953), "Azm & Saade Foundation", and the Lebanese University.

References

[1] Microsoft, "AT&T and Microsoft announce a strategic alliance to deliver innovation with cloud, AI and 5G," https://microsoft.com, Jul 2019.
[2] George Glass, "Telcos will embrace public cloud – they can't afford not to," https://inform.tmforum.org, Sep 2019.
[3] CSA and McAfee, "WP custom apps IaaS trends," Tech. Rep., 2017.
[4] Gartner, "Gartner forecasts worldwide public cloud revenue to grow," https://gartner.com, 2-April-2019.
[5] Ovum, "Understanding the Business Value of Re-architecting Core Applications on the Public Cloud," https://ovum.informa.com, Feb 2019.
[6] McKinsey, "Creating value with the cloud," https://www.mckinsey.com/.
[7] ETSI, "Policy and Charging Control Framework for the 5G System; Stage 2 (3GPP TS 23.503 v15.2.0 Rel. 15)," Jul 2018.
[8] 3GPP, "Network Data Analytics Services; Stage 3 (3GPP TS 29.520 v15.0.0 Rel. 15)," Jul 2018.
[9] G. Liu and H. Shen, "Minimum-cost cloud storage service across multiple cloud," *IEEE/ACM Transactions on Networking (TON)*, 2017.
[10] Z. Wu, M. Butkiewicz, D. Perkins, E. Katz-Bassett, and H. V. Madhyastha, "Spanstore: Cost-effective geo-replicated storage spanning multiple cloud," in *24th ACM Symposium on Operating Systems*, 2013.
[11] Y. Ran, J. Yang, S. Zhang, and H. Xi, "Dynamic IaaS computing resource provisioning strategy," *IEEE Transactions on Services Computing*, 2017.
[12] Q. Wang, M. M. Tan, X. Tang, and W. Cai, "Minimizing cost in IaaS clouds via scheduled instance reservation," in *Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference*, 2017.
[13] 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.
[14] M. Chen, Y. Bao, X. Fu, G. Pu, and T. Wei, "Efficient resource constrained scheduling using parallel two-phase branch-and-bound," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 5, 2017.
[15] P. Avella, M. Boccia, and I. Vasilyev, "A branch-and-cut algorithm for the multilevel generalized assignment," *IEEE Access*, vol. 1, 2013.
[16] M. Y. Lyazidi, L. Giupponi, J. Mangues-Bafalluy, N. Aitsaadi, and R. Langar, "A Novel Optimization Framework for C-RAN BBU Selection," in *IEEE 86th Vehicular Technology Conference (VTC-Fall)*, 2017.
[17] N. Alliance, "NGMN optimised backhaul requirements," *Next Generation Mobile Networks Alliance*, p. 19, 2008.
[18] M. Murthy, H. Sanjay, and J. Ashwini, "Pricing models and pricing schemes of IaaS providers," in *International Conference on Advances in Computing, Communications and Informatics*. ACM, 2012.
[19] 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.
[20] L. Nirenberg, "The Weyl and Minkowski problems in differential geometry in the large," *Communications on pure and applied mathematics*, vol. 6, no. 3, 1953.
[21] G. Appa, "Dantzig-wolfe decomposition algorithm," *Operational Research Quarterly*, vol. 20, no. 2, p. 275, 1969.
[22] IBM, "ILOG CPLEX Optimization Studio," https://ibm.com/analytics/cplex-optimizer/.
[23] China Mobile Research Institute, "C-RAN, The road towards green RAN, v3.0," Apr 2013.
[24] Google, "GCP Pricing," https://cloud.google.com/compute/pricing.
[25] GCP, "Considerations when choosing a VM," https://cloud.google.com/datalab/docs/how-to/machine-type.