HAL
open science

Optimization of Network Services Embedding Costs over Public and Private Clouds

Cedric Morin, Géraldine Texier, Christelle Caillouet, Gilles Desmangles, Cao-Thanh Phan

To cite this version:

Cedric Morin, Géraldine Texier, Christelle Caillouet, Gilles Desmangles, Cao-Thanh Phan. Optimization of Network Services Embedding Costs over Public and Private Clouds. ICOIN 2020 - 34th International Conference on Information Networking, Jan 2020, Barcelone, Spain. 10.1109/ICOIN48656.2020.9016607 . hal-02440297

HAL Id: hal-02440297

https://hal.science/hal-02440297

Submitted on 15 Jan 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Optimization of Network Services Embedding Costs over Public and Private Clouds

Cedric Morin*†‡, Géraldine Texier *†, Christelle Caillouet§, Gilles Desmangles ‡, Cao-Thanh Phan †
*IMT Atlantique/IRISA/Adopnet, France; first.lastname@imt-atlantique.fr
†BCOM, France; first.lastname@b-com.com
‡TDF, France; first.lastname@tdf.fr
§Université Côte d'Azur/I3S/CNRS/Inria, France; first.lastname@univ-cotedazur.fr

***Abstract*—Public cloud providers offer individuals and businesses the ability to rent IT resources to meet their needs without investing in their own hardware. At the same time, the Network Functions Virtualisation (NFV) concept promotes the migration of network operators from expensive and poorly scalable hardware network devices to virtual softwarized network functions. In order to embed those functions, network operators may decide to subscribe to public cloud offers. However, their diversity, both in terms of resource capacity and price, makes it difficult to find the optimal combination of offers that meets all needs at the lowest cost. In this paper we propose to solve this issue with an algorithm designed to help a network operator to select the best combination of offers (in terms of price) to reserve the virtual machines needed to support a set of network services. We analyze the computation time of our solution against various metrics, and estimate the cost savings compared to a traditional resource provision scheme or an unplanned resource rental strategy. Finally we evaluate the opportunity for a network operator to build its own datacenter, considering the existence of offers from public clouds.**

***Index Terms*—NFV, MANO, cloud offers, linear program**

I. Introduction

Recently, the NFV concept [1] appeared to cope with the emergence of 5G, the expected explosion in traffic and service needs, and the softwarization of networks. It consists in transforming network services (generally performed with middleboxes) into software running on generic servers, paving the way for automated and dynamic network management. 5G introduces the concept of slicing to create virtualized network infrastructures (slices), isolated and adapted to specific Network Service (NS) needs, above a physical infrastructure. This emphasized the need for automated management of network service virtualization. ETSI proposed the NFV-Management and Orchestration (MANO) framework [2], an architecture dedicated to manage the Virtual Network Functions (VNFs). Upon reception of a NS creation request, the Network Functions Virtualisation Orchestrator (NFVO) decides how to place the different VNFs that compose the NS, in order to respect the NS's performance constraints. This problem, known as the Virtual Network Function Graph Placement Problem (VNFGPP) (although other names can be encountered), is the subject of many proposals. The placement choice is driven both by the resource offers advertised to the NFVO by Virtualized Infrastructure Managers (VIMs) in charge of the management of virtual infrastructures composed of servers and datacenters and by the QoS metrics of the networks that interconnect them.

Once the VNFs are assigned to their VIMs, the NFVO must still select commercial offers to access resources and embed the sub-components of the VNFs, called Virtual Network Function Components (VNFCs). One VNFC corresponds to exactly one Virtual Machine (VM) running on a server. The VIM announces multiple offers from the cloud operator, or from several cloud operators when the VIM is a broker. The substantial number of offers makes the selection problem more complex. In addition, the NFVO must plan its resource consumption over a long period of time in order to benefit from lower long-term reservation rates.

Main cloud operators include Amazon Web Service (AWS), Microsoft Azure or Google Cloud. Although the details of their offers may vary, two main systems can be observed: guaranteed and spot. Clients choosing guaranteed offers have to pay a fix known price to use the resources. Once the price is paid the resources are granted. Regarding spot offers, a varying hidden reserve price is defined for each type of resource by the cloud operator. This price can be higher or lower than the guaranteed price for the same resource. Clients must bid on the price they are willing to pay for the resources, those who make an offer higher than the reserve are granted the resource for the time period, the others have to wait. Since running instances may be interrupted, spot instances are recommended for delay tolerant jobs only, which excludes most of the 5G use cases. Consequently in this paper we only consider guaranteed offers (detailed in Section III).

In this article, we propose an algorithm designed to help a network operator's NFVO to select the best combination of offers (in terms of price) to reserve the VMs needed to support a set of NS. The algorithm's inputs consist in public offers from different cloud computing providers, as well as traffic load forecasts and price estimates for the following year from the network operator. To the best of our knowledge, this is the first paper to address this issue in this context. Such an algorithm allows a network operator to plan its expenses over the next period and pay resource reservations in advance to lower the costs. In addition, we show that it can be used by a network operator to evaluate the utilization rate of a possible future private datacenter.

The paper is organized as follows: Section II provides a

brief overview of different works related to cost optimization at different levels of the MANO framework. Section III introduces our model while Section IV presents both its evaluation in terms of runtime and cost, and how it can be used to assess the relevance for a network operator to invest in a private cloud. We conclude in Section V.

II. Related works

To analyse the scope of the different contributions proposed in the literature, we recall the definition of the three actors involved. The cloud provider, represented by the VIM in the MANO framework, sells virtual resources, such as VMs. The NFVO uses those VMs to build consistent network services with given features, guaranteed capacity and QoS. The slice provider consumes those NSs to create a slice, a virtual network with a set of embedded NSs that can address a family of use cases.

Several papers are analyzing the best pricing scheme to maximize profits from the cloud provider perspective. They mainly consider the spot offer because its bidding mechanism is subject to many optimizations. [3] focuses exclusively on spot instances, and aims to optimize cloud provider revenues. Authors in [4] have a similar objective. They are interested in hybrid pricing schemes that present both spot and guaranteed offers. Using game theory and queuing theory they demonstrate that, in most of the cases, spot offers should not be proposed. Moreover, they show that only a waiting cost threshold determines whether or not a job will try to bid on the spot instance market. In our context of a network operator, a network function that is not started in due time looses all its value, and potentially discontents many customers. Consequently, this property supports our choice to ignore spot instances. In the context of public clouds with apparent infinite resources, authors in [5] show that a provider may artificially simulate a shortage to maximize their profit. They briefly point out that, for long term jobs, cloud consumers (like NFVOs) should consider reserved options, or even buying their own hardware, but they do not perform any further analysis.

From the NFVO perspective, the most explored way to reduce costs is through VNFGPP [6], [7], [8], [9], [10]. Although the techniques, context and side objectives may differ, the strategies of those papers follow a similar pattern: in order to reduce the resource cost they consolidate the placement, using each instantiated VNF at the maximum of its capacity. This process is purely internal to the NFVO and does not involve any other actor. Authors in [11] adopt another strategy that does not follow MANO architecture, so it is difficult to apply their approach to our situation. However, if we map what they refer to as a "chunk of network" onto a NS, we can use their proposal to address the case where the slice provider has a set of slices to instantiate, and the NFVO disposes of a limited set of NSs. The objective is to maximize the profit of the NFVO using successively price competition, auction and optimisation.

All of those approaches consider interesting and complementary solutions to lower the costs or optimize the revenue for the different actors of the architecture. However, to the best of our knowledge no publication so far investigated the choice that the NFVO has to make when facing different types of commercial offers to buy resources in order to actually deploy the VNFCs requested to run the NSs.

In this paper we propose an Integer Linear Program (ILP) that addresses this challenge. We evaluate its computation time against various parameters and we provide cost comparisons of our algorithm with baseline solutions. Finally we suggest that this system can assist the network operator in determining the opportunity to build its own private datacenter.

III. Model and problem description

In this problem, we suppose that the NS provider has a prevision of the traffic he will have to manage for a given period of time in the future, such as one year, based on past experience. From this estimation, he can deduce the NS needed to handle the traffic, and place them into the network using any version of the VNFGPP, as the ones presented in Section II. This placement takes care of all QoS related constraints, such as delay, loss, or amount of available bandwidth along the path. Once every VNFC is assigned to a VIM for any time of the foreseen period, the NFVO can start to evaluate which offers should be selected from each VIM individually. It is this last step that we handle in this paper.

A. Offers description

VIM offers can take many different forms and it would be impossible to consider them all. We first describe the most generic offer possible, then we detail how to express the offers of AWS, one of the leaders of the cloud business, with this template. An offer proposed by a cloud operator represents the pricing of a specific VM template, referred to as flavor, over a given time interval (*i.e.* one or more time slots). Once an offer is paid, the VM of the corresponding flavor can be instantiated during the given time slots. A flavor may have a variety of attributes. Without loss of generality we consider only CPU and RAM. Similarly, we suppose that VNFCs only require CPU and RAM to run. Each flavor may be instantiated using different reservation offers. As evoked in Section I, we only consider guaranteed offers. The most generic form of reservation is composed of a fixed cost paid in advance, a variable cost paid on a per-use basis and a set of time slots that defines when the reservation can be used. The variable cost might be subject to market fluctuations, and we suppose that the network operator keeps a record of those prices to be able to predict their future variations. We emphasize that an offer is bounded to a specific flavor and, once paid, cannot be used for another one. The flavor is bounded to a cloud operator (although different cloud operators can happen to deliver the same type of flavor). Our model is designed to handle such a generic reservation offer, but we can easily derive more specific cloud operator related offers. As an example we introduce some AWS tariffs translated in our framework[1]:

[1] AWS pricing, https://aws.amazon.com/ec2/pricing/

- On demand offers are purely "pay per use": no time slot restriction and no fixed cost.
- Reserved offers have a limited duration (1 or 3 years) and an upfront cost, but the variable cost is null.
- Scheduled offers are similar to reserved offers, except that they apply only on given hours within the day. The equivalent hour rate is the same as for reserved offers.

Our algorithm outputs the amount of each offer that should be used at each time slot of the future period.

B. Notations

Typically a VNFC will keep running during multiple time slots. From the NS consumer point of view it represents one unique VNFC. In our model however, a VNFC is bounded to one unique time slot. Consequently, one VNFC running during N time slots is represented by N VNFCs, each one running during one of the N considered time slots.

We remind that, following MANO norm, a VNFC can be hosted by one VM only (it cannot be splitted over multiple VMs), and one VM can host at most one VNFC at a time even if this VNFC does not consume all the resources.

Notations used in the model are summarized in Table I. In the whole model "operator" refers to a cloud operator.

TABLE I: **Notations**

Name	Description
V	Set of VNFCs to be placed
T	Set of time slots (of equal length)
V_t	Set of VNFCs to be placed during time slot t
t_v	Time slot $t \in T$ during which VNFC $v \in V$ has to run
C^v	CPU requested by the VNFC $v \in V$
S^v	Storage (RAM) requested by the VNFC $v \in V$
O	Set of operators
F_o	Set of flavors proposed by $o \in O$
R_{f_o}	Available reservations for flavor $f_o \in F_o$
C_{f_o}	CPU of flavor $f_o \in F_o$
S_{f_o}	Storage (RAM) of flavor $f_o \in F_o$
C_o	Total amount of CPU that operator $o \in O$ can provide
S_o	Total amount of storage that operator $o \in O$ can provide
$p_{o,f,r,t}$	Variable price for reservation $r \in R_{f_o}$ for time slot $t \in T$
$P_{o,f,r}$	Fixed price for reservation $r \in R_{f_o}$

C. ILP model

1) Variables: We introduce three types of variables representing the VM instances (1), the VNFC embedding on these instances (2) and the fixed costs that have to be paid (3). A triplet $[o, f, r]$ defines a reservation of flavor f provided by operator o under reservation tariff r.

$$\phi_{o,f,r,t} = 0..|V_t| \quad \text{Number of VMs } [o,f,r] \text{ to reserve at time t} \tag{1}$$

$$x^v_{o,f,r,t} = \begin{cases} 1 & \text{if VNFC } v \in V \text{ is installed on a VM obtained} \\ & \text{through reservation } [o,f,r] \text{ at time slot } t \in T_v \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$\Phi_{o,f,r} = 0..\max_{t \in T}(|V_t|) \quad \text{Number of reservation } [o,f,r] \text{ paid} \tag{3}$$

2) Objective: The objective of the problem is to minimize the expected cost over the full period. The total cost includes the payment of variable prices only when a VM is actually running, and of fixed prices for the long-term reservations.

$$\min \sum_{o \in O} \sum_{f \in F_o} \sum_{r \in R_{f_o}} \left(\sum_{t \in T} \phi_{o,f,r,t} * p_{o,f,r,t} + \Phi_{o,f,r} * P_{o,f,r} \right) \tag{4}$$

3) Constraints: Each VNFC must be instantiated during its time slot (5)

$$\sum_{o \in O} \sum_{f \in F_o} \sum_{r \in R_{f_o}} x^v_{o,r,f,t_v} = 1 \ \forall \ v \in V \tag{5}$$

A VNFC can be installed on a VM only if the corresponding flavor has enough CPU (6) and RAM (7) to host it :

$$x^v_{o,r,f,t_v} * C^v \leq x^v_{o,r,f,t_v} * C_f \ \forall \ (o,r,f,v) \in (O,R,F,V) \tag{6}$$

$$x^v_{o,r,f,t_v} * S^v \leq x^v_{o,r,f,t_v} * S_f \ \forall \ (o,r,f,v) \in (O,R,F,V) \tag{7}$$

Since each VNFC has to get its own VM their should be at least as much instances as VNFCs (8) :

$$\sum_{v \in V_t} x^v_{o,r,f,t_v} \leq \phi_{o,f,r,t} \ \forall \ (o,r,f,v) \in (O,R,F,V) \tag{8}$$

An operator cannot offer more CPU (9) or RAM (10) than its capacity.

$$\sum_{f \in F_o} \sum_{r \in R_{f_o}} \phi_{o,f,r,t} * C_f \leq C_o \ \forall \ (t,o) \in (T,O) \tag{9}$$

$$\sum_{f \in F_o} \sum_{r \in R_{f_o}} \phi_{o,f,r,t} * S_f \leq S_o \ \forall \ (t,o) \in (T,O) \tag{10}$$

At any time there must be at least as many fixed costs paid as reservations used (11)

$$\phi_{o,f,r,t} \leq \Phi_{o,f,r} \ \forall \ (t,o,f,r) \in (T,O,F,R) \tag{11}$$

D. The licence problem

In addition to the VMs, the cost of the VNFs licences is another important aspect that must be taken into account when evaluating the total cost of a service. Reducing it implies to know the licence billing method. This is actually non trivial, since many systems exist and are not normalized [12].

The papers presented in Section II that tackle this issue use the VNFGPP to try to minimize licence cost and the resource cost all together using consolidation. This logic is directly inspired from traditional networks with physical middleboxes. Middleboxes, just as their licences, should be used at full potential, else they are partially wasted. VNFs, however, have a major property: they can scale up or down, depending on the traffic load, which may reduce considerably the interest of consolidation. Just as licence billing system followed traditional middelboxes logic of exploitation, we may suggest that VNF licence could gradually embrace VNF work flows. Some major actors already issued "pay per use" licences

similar to AWS on demand offers [13], and in the future they may produce more complex offers to mirror the ones proposed today for VM reservations. If so, our model could be used to handle licence reservations as well.

IV. Simulations and results

We used the Gurobi solver [14], 4 logical cores Intel i5-6200U and 2GB of RAM to evaluate our algorithm.

A. Parameters

1) Incoming traffic: To simulate the traffic we use the dataset of the City of Milano provided in [15]. For the sake of the example we decide to focus on the deployment of one specific VNF representing the Mobility Management Entity (MME). We extract the callIn and callOut activities from the dataset (which corresponds to actions that would trigger the MME) and convert this activity into a number of MMEs that must be provisioned to handle it, based on the work presented in [16]: one MME of 2 CPU and 2 GB of RAM to handle 40 requests. As a result we obtain the required number of MMEs by time slot of 10 minutes for November 2013 (represented in Figure 2), December 2013 and the 1st of January 2014. We can note that the traffic is cyclic and follows day/night shifts. Here we suppose that the MME VNF is composed of only one VNFC, which may not be the case in practice but clarifies our demonstrations. We also consider that the MME only scales out and do not scales in. This is a strategic choice that has to be made beforehand by the NFVO, our algorithm adapts to the choice by selecting different flavors. We consider that all the MMEs are allocated to the same VIM for the whole city, else the algorithm should be run separately for each VIM.

2) Offers: Regarding cloud offers, we consider two operators: AWS public cloud and a private cloud operated by the network operator itself. Since we do only have one type of VNFC to host, only one flavor will be proposed. For this example we chose AWS m5.large (2 CPU 8 RAM) instances, adapted to general purpose computing.

We assume here that, from a client point of view, public cloud resources are infinite. This assumption is consistent with other approaches on the subject [4][5][13]. As a consequence, AWS has enough resources to host all our VNFCs anytime. We then selected the 3 main AWS offers and their respective tariffs taken from AWS website on the 22/05/2019. On Demand (OD) : 0.096$ per hour, no fixed cost. Reserved 1 year (R) : 0$ per hour, 501$ fixed cost. Scheduled daily (S) from 8h00 to 20h00 (daytime) : 0$ per hour, 250.5$ fixed cost.

The private cloud, on the contrary, has limited resources. Although it has no reason to be very small, since we consider only one VNF among all the VNFs that a network operator should deploy we only grant it a fraction of classical capacities: we consider it has 20 CPUs and 80 RAMs. Since it belongs to the network operator, the hourly cost should be marginal, corresponding only to the extra electricity consumption. However, to better estimate the real cost of using the datacenter we estimated the OPEX cost taking into consideration hardware, staff and electricity costs. We obtained an approximated hourly rate of 0.012$ (P offer).

B. Runtime

We analyze the ILP runtime regarding the number of available offers, the length of the foreseen period and the number of VNFCs to place. Results are presented in Figure 1.

1) Length of the period: To produce different period lengths we take one, two, and three weeks of November, the full month, and finally we concatenated from two to twelve times November to obtain the equivalent of a year. We propose two offers to the algorithm: R and OD. In Figure 1a, we observe that the computation time grows linearly with the length of the period. While other parameters can increase a lot, the length of the period is bounded. Indeed, even if AWS proposes 3 years offers, the precision of the traffic prediction and offer price variation will decrease over time, making long term commitments hazardous.

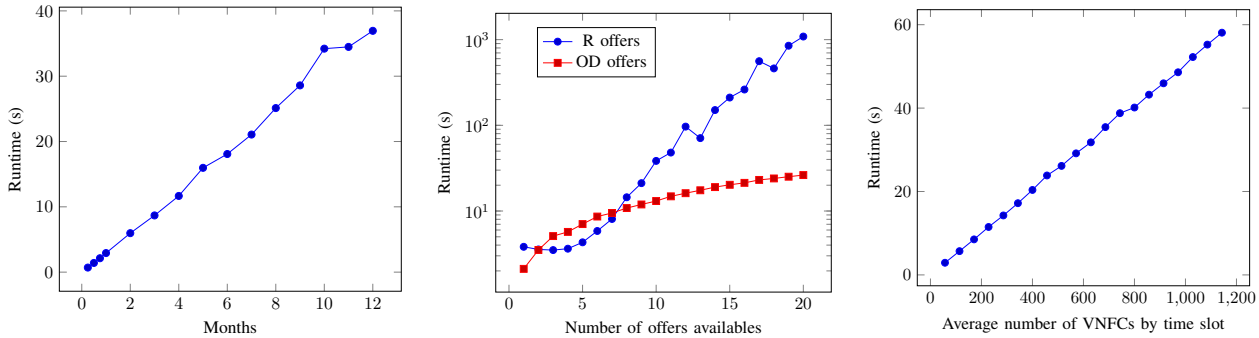
2) Number of offers available: From a complexity point of view, introducing more cloud operators, more flavors or more offers is equivalent. Therefore, we choose to simply focus on the multiplication of offers in this section. Because R and OD offers have very distinct characteristics we analyze them separately and display the results in Figure 1b. Regarding OD offers, we build an offer by randomly taking, for each time slot, a price between 0.060$/h and 0.180$/h, and we propose from 1 to 20 offers. For R offers, we propose the classical OD and R offers, plus between 1 and 20 S offers. S offers have a duration of 4 hours, and start every hour: with 20 of them the full day is covered. We observe that the computation time is linear with the number of OD offers, but exponential with R ones. For the first R offers the computation is constant, this is because the first proposed R offers apply before the daily traffic peak, so they are not interesting compared to R and OD and the ILP quickly discards them.

The exponential complexity sets a limit to the capabilities of our model: all possible S offers cannot be considered, a choice has to be made. In our use case, the daily periodicity of the work load makes this task relatively easy, and selecting simply one S offers out of all the possible ones greatly improves the overall cost (see next section).

3) Average number of VNFCs by time slot: We also study the behavior of the ILP over the month of November proposing only R and OD offers, and we multiply the number of VNFC to place at each time slot by 1 (initial situation) up to 20. We observe in Figure 1c that the computational time grows linearly with the number of VNFCs, which allows our ILP to be used in large size datacenters on which the network operator may have several VNFCs to embed.

C. Cost

We evaluate the interest for a network operator to use our algorithm by comparing different reservation strategies. We first suppose that the network operator does not have a private datacenter. When operating their own systems, network operators tend to dimension them not to absorb the average traffic load, but rather to handle peak loads [13], which leads to overprovisioning. Translated directly into AWS language, it would mean taking only R offers. We refer to it as the



(a) Runtime by length of the period (b) Runtime by number of offers (R or OD) (c) Runtime by number of VNFCs to place

Fig. 1: Comparison of ILP runtime with different parameters

R strategy. Even if this strategy doesn't seem complicated, choosing an optimal set of reservations when multiple sizes of VNFCs are present already requires some planning. Taking advantage of the flexibility of the cloud, the network operator may decide to buy only OD offers to face the traffic as it comes, using a straightforward OD strategy without any further planning. It could also decide to mix R and OD offers (the $OD + R$ strategy). Lastly, noticing that the traffic strongly follows the night and day cycle, it could opt for a scheduled daytime offer (the $OD + R + S$ strategy). This strategy produces the optimal cost provided by the public cloud, given the offer we chose to focus on, and we base the comparison with all other costs on it (see Figure 3). The network operator may wonder what would be the final cost if he was owning its own private datacenter. To give an answer, we introduce an $OD+R+S+P$ strategy considering OD, R, S offers plus a P offer corresponding to the placement in its private datacenter. The result of this last strategy is displayed in Figure 2. We can note that the R offer is actually not used because the S offer fits much better the workload needs, and the reduced nightly traffic is absorbed by the P offer.

The comparison of the different strategies' costs is provided in Figure 3. First of all, we can note that the R strategy performs very bad. This is due to the 1st of January traffic peak that forces the network operator to book a handful of resources that will stay idle the rest of the time. This reflects well the default of the classical overprovisioning strategy. Second, the difference between OD and $OD + R$ is quite small due to the very low traffic at night, which makes reservations quite unattractive. The $OD + R + S$ strategy performs very well because S focuses precisely on peak hours. Thus, using our algorithm to plan in advance the offer selection can bring significant advantages compared to R and OD strategies.

D. Private datacenter utilization

In this section we focus on the situation where a network operator has to decide whether or not to build its own datacenter to absorb a portion of the traffic. In Section IV-C we estimated the benefits in term of costs, however this does not take into account the global investment required to build

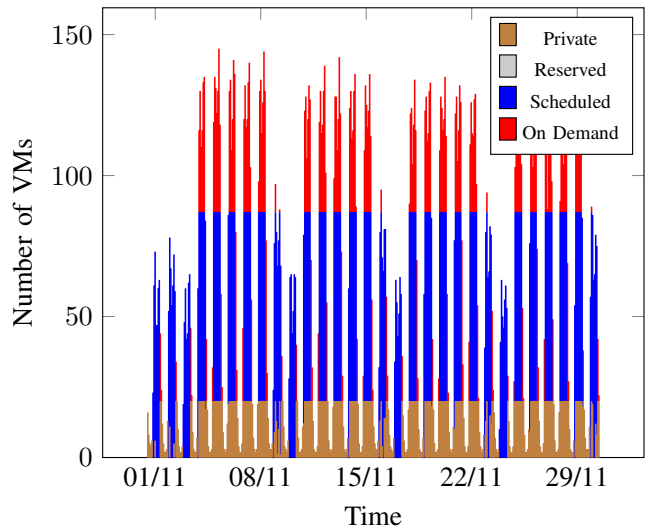


Fig. 2: Selected offers through time

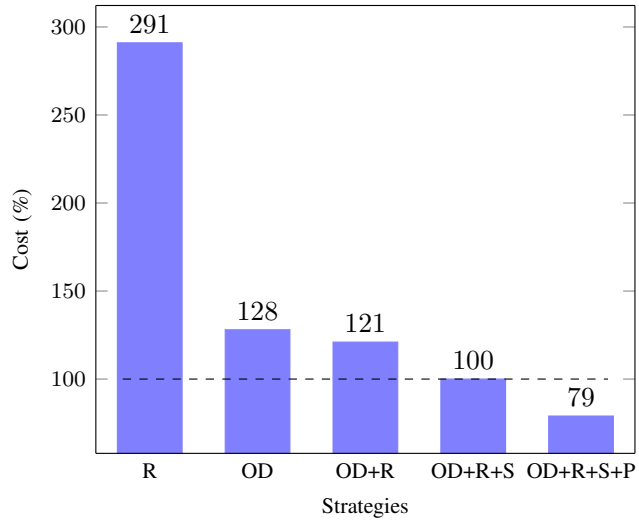


Fig. 3: Cost performance using different offers

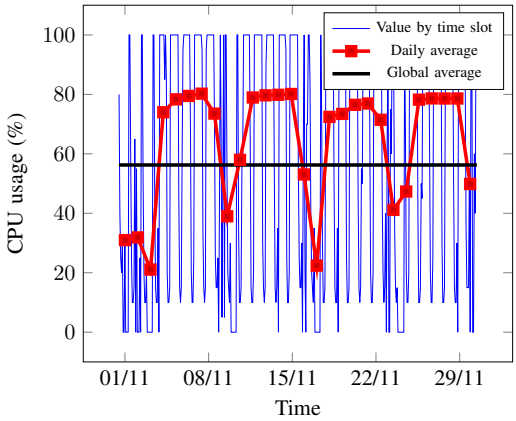


Fig. 4: Private cloud CPU utilization through time

the infrastructure. Since such a facility is costly to build, we suppose that a network operator will be interested in knowing whether it will be used at the maximum of its capacity or not, and maybe in predicting the periods when it can sell the unused resources. To analyze this, we take the results of Section IV-C provided by strategy $OD + R + S + P$, and we measure the amount of CPU used at any moment. Results over the month of November are presented in Figure 4.

As expected, the average utilization rate is not 100% since the traffic at night is very low: the utilization rate is only around 70% - 80% during the week days. It is even lower during the week-end: although the traffic is lower, we could expect the datacenter to be fully used during the day. However, since scheduled offers are reserved for the full week and cannot be cancelled for the week-end, it turns out to be more economic to use the already-paid resources rather than the private datacenter ones. This effect is especially important during Sundays, when the traffic is at its lowest level (see Figure 2). This affects the global utilization rate that drops at 56%. We conclude that taking into consideration available commercial offers, especially regarding periodic traffic, can modify a lot the actual benefit a network operator can expect from building its own facilities.

V. Conclusion and perspectives

In this paper, we proposed a model to assist the NFVO in the process of selecting cloud providers offers, in order for it to buy enough resources to embed all the required VNFCs at the best possible price in due time. Based on NFVO's workload predictions, our model allows to plan in advance long-term reservations, which come with reduced hourly prices. We applied this technique to a network operator use-case using a real dataset. We showed that the model keeps doing well when the amount of VNFCs, On Demand offers and length of the prediction increased. The long-term scheduled reservation offers however induce exponential complexity. We mitigated this shortcoming by taking advantage of the periodicity of the traffic, showing that selecting only one well-chosen scheduled offer already fairly reduced the overall placement cost. Finally, we stressed the fact that taking into account existing commercial offers is really important for a network operator when it comes to decide whether or not to build its own private cloud. Although the operational expenditure (OPEX) is reduced, the capital expenditure (CAPEX) may not be as worthy as expected since the projected utilization rate might be lower than anticipated.

For future work, we plan to develop an heuristic to handle the exponential complexity induced by reserved offers. We would also like to propose an algorithm dedicated to actually assign a specific VM to each VNFC, minimizing the migrations between VMs, which is the next and last step the NFVO has to perform to fully embed a network service.

Acknowledgement

This work was partly funded by the project H2020-ICT-2016-2 "5G-TRANSFORMER" (761536).

References

[1] "Network Functions Virtualisation, An Introduction, Benefits, Enablers, Challenges & Call for Action." Darmstadt-Germany: ETSI, Oct. 2012.

[2] ETSI GS NFV-MAN 001, "Network Functions Virtualisation (NFV); Management and Orchestration," 12 2014, version 1.1.1.

[3] J. Song and R. Guerin, "Pricing and bidding strategies for cloud computing spot instances," in 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). Atlanta, GA: IEEE, May 2017, pp. 647–653.

[4] V. Abhishek, I. Kash, and P. Key, "Fixed and Market Pricing for Cloud Services," Proc. NetEcon'12, Mar. 2012.

[5] C. Kilcioglu and C. Maglaras, "Revenue Maximization for Cloud Computing Services," SIGMETRICS, Nov. 2015.

[6] B. Addis, G. Carello, F. D. Bettin, and M. Gao, "On a Virtual Network Function Placement and Routing problem: properties and formulations," p. 39, Nov. 2018.

[7] S. Lange, A. Grigorjew, T. Zinner, P. Tran-Gia, and M. Jarschel, "A Multi-objective Heuristic for the Optimization of Virtual Network Function Chain Placement." IEEE, Sep. 2017, pp. 152–160.

[8] O. Soualah, M. Mechtri, C. Ghribi, and D. Zeghlache, "Energy Efficient Algorithm for VNF Placement and Chaining," in 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID). Madrid, Spain: IEEE, May 2017, pp. 579–588.

[9] M. C. Luizelli, W. L. da Costa Cordeiro, L. S. Buriol, and L. P. Gaspary, "A fix-and-optimize approach for efficient and large scale virtual network function placement and chaining," Computer Communications, vol. 102, pp. 67–77, Apr. 2017.

[10] Y. Sang, B. Ji, G. R. Gupta, X. Du, and L. Ye, "Provably efficient algorithms for joint placement and allocation of virtual network functions," in INFOCOM 2017-IEEE Conference on Computer Communications, IEEE. IEEE, 2017, pp. 1–9.

[11] M. Jiang, M. Condoluci, and T. Mahmoodi, "Network slicing in 5g: An auction-based model," in 2017 IEEE International Conference on Communications (ICC). Paris, France: IEEE, May 2017, pp. 1–6.

[12] ETSI GR NFV-EVE 010, "Licensing Management; Report on License Management for NFV," 12 2017, version 3.1.1.

[13] M. Armbrust, I. Stoica, M. Zaharia, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, and A. Rabkin, "A view of cloud computing," Communications of the ACM, vol. 53, no. 4, p. 50, Apr. 2010.

[14] L. Gurobi Optimization, "Gurobi optimizer reference manual," 2018. [Online]. Available: http://www.gurobi.com

[15] G. Barlacchi, M. De Nadai, R. Larcher, A. Casella, C. Chitic, G. Torrisi, F. Antonelli, A. Vespignani, A. Pentland, and B. Lepri, "A multi-source dataset of urban life in the city of Milan and the Province of Trentino," Scientific Data, vol. 2, no. 1, p. 150055, Dec. 2015.

[16] I. Alawe, A. Ksentini, Y. Hadjadj-Aoul, P. Bertin, and A. Kerbellec, "On evaluating different trends for virtualized and SDN-ready mobile network," in 2017 IEEE 6th International Conference on Cloud Networking (CloudNet). Prague, Czech Republic: IEEE, Sep. 2017, pp. 1–6.