



HAL
open science

Optimisation du coût de déploiement de services réseau virtualisés dans le cloud

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. Optimisation du coût de déploiement de services réseau virtualisés dans le cloud. CORES 2020 – 5ème Rencontres Francophones sur la Conception de Protocoles, l'Évaluation de Performance et l'Expérimentation des Réseaux de Communication, Sep 2020, Lyon, France. hal-02874121

HAL Id: hal-02874121

<https://hal.science/hal-02874121>

Submitted on 18 Jun 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.

Optimisation du coût de déploiement de services réseau virtualisés dans le cloud

Cédric Morin¹²³ et Geraldine Texier¹² et Christelle Caillouet⁴ et Gilles Desmangles³ et Cao-Thanh Phan²

¹IMT Atlantique/IRISA/Adopnet, France

²BCOM, France

³TDF, France

⁴Université Côte d'Azur/I3S/CNRS/Inria, France

Les opérateurs de clouds publics offrent aux particuliers et aux entreprises la possibilité de louer des ressources informatiques pour répondre à leurs besoins sans investir dans leur propre matériel. Le concept de Network Functions Virtualisation (NFV) favorise la migration des opérateurs réseau de dispositifs matériels dédiés coûteux et peu évolutifs vers des fonctions réseau logicielles virtualisées. Afin de déployer ces fonctions, les opérateurs réseau peuvent décider de s'abonner à des offres de clouds publics. Toutefois, leur diversité, tant en termes de capacité de ressources que de prix, fait qu'il est difficile de trouver la combinaison d'offres répondant à tous les besoins au moindre coût. Nous proposons de résoudre ce problème à l'aide d'un programme linéaire conçu pour aider un opérateur réseau à sélectionner la meilleure combinaison d'offres (en termes de prix) permettant de réserver les machines virtuelles nécessaires pour supporter un ensemble de services réseau. Nous analysons le temps d'exécution de notre solution, et nous estimons les économies réalisées par rapport à un schéma traditionnel de mise à disposition de ressources ou à une stratégie de location de ressources sans planification. Enfin, nous évaluons l'opportunité pour un opérateur réseau de construire son propre datacenter alors que des offres de clouds publics sont disponibles.

Mots-clefs : NFV, MANO, offres cloud, programme linéaire

1 Introduction

Le concept de NFV consiste à transformer les Network Services (NSs), habituellement implémentés sur du matériel physique dédié, en fonctions logicielles pouvant être exécutées sur des serveurs génériques. Suivant le modèle ETSI [ETS14], ces NSs sont composés de graphes de Virtual Network Functions (VNFs), qui représentent des fonctions réseau unitaires (*ex.* : firewall, encoder ...). Elles sont elles-mêmes décomposées en graphes de Virtual Network Function Components (VNFCs), qui représentent des entités unitaires du point de vue des ressources, chacune hébergée sur un unique Virtual Deployment Unit (VDU). Un VDU est un ensemble de ressources accessibles par un client, par exemple une Machine Virtuelle (VM), et qui ne peut accueillir qu'un seul VNFC. Lorsqu'une demande de création de NS est émise, le Network Functions Virtualisation Orchestrator (NFVO) est en charge de placer la chaîne de composants correspondante dans le réseau. Pour cela il s'appuie sur la vision du réseau proposée par les Virtualized Infrastructure Managers (VIMs). Elle est composée de nœuds représentant chacun un Point of Presence (PoP) (typiquement un datacenter), reliés par des liens exposant une certaine bande passante, un délai et un taux de pertes. Le placement de NSs dans le réseau en fonction de toutes ces contraintes est un problème connu sous le nom de Virtual Network Function Graph Placement Problem (VNF-GPP).

Dans cet article nous nous intéressons à l'étape directement après ce placement : une fois les différents PoPs sélectionnés il faut encore avoir effectivement accès aux ressources exposées, ce qui nécessite de passer un contrat avec l'opérateur cloud qui en est le propriétaire. Nous proposons un modèle destiné à aider le NFVO à sélectionner la meilleure combinaison d'offres (en termes de prix) permettant d'obtenir assez de VMs pour supporter les services demandés à tout instant. Cet article est un résumé de [MTC⁺20].

2 Le modèle

TABLE 1: Notations

Nom	Description	Nom	Description
O	Ensemble des opérateurs	C_o	Quantité de CPU dont dispose $o \in O$
T	Ensemble des créneaux horaires	S_o	Quantité de RAM dont dispose $o \in O$
t_v	Créneau horaire $t \in T$ où $v \in V$ s'exécute	F_o	Ensemble des flavors proposées par $o \in O$
V	Ensemble des VNFCs à placer	R_{f_o}	Réservations disponibles pour $f_o \in F_o$
V_t	Ensemble des VNFCs à placer durant $t \in T$	C_{f_o}	CPU proposé par $f_o \in F_o$
C^v	Besoin en CPU de $v \in V$	S_{f_o}	RAM proposée par $f_o \in F_o$
S^v	Besoin en RAM de $v \in V$	$P_{o,f,r,t}$	Prix de la réservation $r \in R_{f_o}$ pour $t \in T$
$P_{o,f,r,\tau}$	Prix de la réservation $r \in R_{f_o}$ sur la plage horaire $\tau \subseteq T$		

Nous établissons ici un modèle général en nous basant sur les offres proposées par Amazon Web Service (AWS)[†], l'un des principaux opérateurs cloud. Les notations utilisées dans cette section sont récapitulées Table 1. Les offres commerciales pour acheter des ressources peuvent potentiellement être très variées, tant sur la forme concrète des ressources que sur la méthode de tarification. Chaque opérateur $o \in O$ propose un ensemble flavors F_o , qui sont des modèles de VDUs (ici des VMs) ayant une quantité donnée de ressources (du CPU C_{f_o} et de la RAM S_{f_o}). Le prix d'une flavor est divisé en deux composantes, un prix à la demande payé uniquement au moment où la flavor est utilisée, $p_{o,f,r,t}$, et un prix lié à la réservation de cette flavor pour une période de temps donnée τ , $P_{o,f,r,\tau}$ payé en début de période. Pour un créneau horaire $t \subseteq \tau$, si les deux prix sont payés l'opérateur réseau a accès à la VM correspondante. L'opérateur réseau doit anticiper ses besoins futurs pour calibrer les réservations. Cette anticipation se fait sur une période notée T , divisée en intervalles de temps unitaires de tailles égales t . De manière générale, la durée de vie d'un VNFC peut s'étaler sur N intervalles de temps. Du point de vue de l'opérateur réseau il s'agit d'un unique VNFC, cependant nous le modélisons ici de façon transparente comme N VNFCs, chacun installé sur l'un des N intervalles de temps considérés. Notre modèle prend en entrée les prévisions de trafic de l'opérateur pour la période à venir (basées sur son expérience passée) converties en quantité de VNFC V_t à placer à chaque instant t et les prévisions des prix des ressources à la demande pour cette même période (les opérateurs de clouds publics fournissant un historique des prix). Il fournit en sortie le nombre de chaque type de réservation à payer immédiatement $\Phi_{o,f,r,\tau}$, la prédiction du nombre de chaque offre à la demande à payer à chaque intervalle de temps de la période $\phi_{o,f,r,t}$ et un placement possible des VNFCs $x_{o,f,r,t}^v$, la variable valant 1 si le VNFC v est installé à l'instant t sur une VM obtenue via la réservation $[o, f, r]$, 0 sinon.

Notre objectif est de minimiser le coût total sur la période considérée (1), tout en respectant un certain nombre de contraintes. Chaque VNFC doit effectivement pouvoir être installé au moment voulu afin de pouvoir absorber la charge de travail prédite (2). Pour le recevoir, une VM doit disposer d'assez de ressources (CPU (3) et RAM (4)). Puisque chaque VNFC doit disposer de sa propre VM il faut au moins autant de VMs (*i.e.* : de prix à la demande payés) que de VNFCs (5), et un prix à la demande ne peut être accessible que si la réservation qui l'englobe est réglée (6)(7). Pour finir, les opérateurs cloud proposant ces VMs ne peuvent en outre offrir plus de ressources que ce qu'ils possèdent (8)(9).

$$\min \sum_{o \in O} \sum_{f \in F_o} \sum_{r \in R_{f_o}} \left(\sum_{t \in T} \phi_{o,f,r,t} * p_{o,f,r,t} + \Phi_{o,f,r,\tau} * P_{o,f,r,\tau} \right) \quad (1)$$

$$\sum_{o \in O} \sum_{f \in F_o} \sum_{r \in R_{f_o}} x_{o,r,f,t_v}^v = 1 \quad \forall v \in V \quad (2) \quad x_{o,r,f,t_v}^v * C^v \leq x_{o,r,f,t_v}^v * C_f \quad \forall (o, r, f, v) \in (O, R, F, V) \quad (3)$$

$$x_{o,r,f,t_v}^v * S^v \leq x_{o,r,f,t_v}^v * S_f \quad \forall (o, r, f, v) \in (O, R, F, V) \quad (4)$$

$$\sum_{v \in V_t} x_{o,r,f,t_v}^v \leq \phi_{o,f,r,t} \quad \forall (o, r, f, v) \in (O, R, F, V) \quad (5)$$

$$\phi_{o,f,r,t} \leq \Phi_{o,f,r,\tau} \quad \forall (t, o, f, r) \in (\tau, O, F, R) \quad (6) \quad \phi_{o,f,r,t} = 0 \quad \forall (t, o, f, r) \in (T \setminus \tau, O, F, R) \quad (7)$$

$$\sum_{f \in F_o} \sum_{r \in R_{f_o}} \phi_{o,f,r,t} * C_f \leq C_o \quad \forall (t, o) \in (T, O) \quad (8) \quad \sum_{f \in F_o} \sum_{r \in R_{f_o}} \phi_{o,f,r,t} * S_f \leq S_o \quad \forall (t, o) \in (T, O) \quad (9)$$

†. AWS pricing, <https://aws.amazon.com/ec2/pricing/>

3 Expérimentations

Afin de simplifier nos expériences nous analysons la réservation de ressources pour une seule VNF : le Mobility Management Entity (MME). Dans les réseaux 4G, le MME est responsable de l'attachement, du détachement et de la mobilité de l'utilisateur vis à vis du réseau. En tant que fonction de contrôle du coeur de réseau, le MME n'a pas besoin de latence extrêmement faible, mais consomme des données. Il est donc adapté à une migration vers le cloud. Nous supposons que le MME est composé d'un seul VNFC. Concernant le trafic à venir nous nous basons sur le jeu de données de la ville de Milan fourni par [BDNL⁺15] qui couvre les mois de novembre et décembre 2013, ainsi que le 1^{er} janvier 2014, par tranches de 10 minutes. Nous agrégeons les activités CallIn et CallOut du MME, et nous les convertissons en quantité de VMs nécessaires en nous basant sur les travaux de [AKHA⁺17] : 1 MME de 2 CPU et 2 GB de RAM par tranche de 40 requêtes/s. La quantité de VMs requise à chaque instant pour le mois de novembre est présentée Figure 1b. Nous nous basons sur trois offres commerciales d'AWS, avec les tarifs disponibles au 22/05/2019 pour une flavor de 2 CPUs et 8 GB de RAM. À la demande (OD) : 0.096\$/h, pas de coût fixes. Bien que notre modèle estime l'utilisation future de cette offre pour évaluer les coûts totaux, les ressources ne sont en fait achetées qu'au dernier moment, ce qui permet d'adapter la quantité de ressources achetées aux besoins réels. Réservation 1 an (R) : 0\$/h, 501\$ fixes. Programmée (S) de 8h à 20h tous les jours, 0\$/h, 250.5\$ fixes. Les valeurs annuelles sont rapportées à 2 mois, en raison de la taille de notre jeu de données de trafic. Enfin nous considérons l'offre privée (P), correspondant au cas où l'opérateur réseau décide de construire son propre datacenter. En estimant les coûts de matériel, d'électricité et de personnel nous évaluons le coût de cette "offre" à 0.012\$/h. Alors que AWS possède des ressources considérées comme étant illimitées, le datacenter privé ne possède que 20 CPU et 80 GB de RAM. C'est relativement peu, mais notre scénario ne considère que le placement d'une VNF parmi tous les services que doit rendre un opérateur réseau, et nous réduisons la taille du datacenter en conséquence. Nous évaluons notre solution en utilisant le solveur Gurobi et 12 cœurs logiques Intel Xeon E5-2630.

Nous commençons par analyser les bénéfices qu'un opérateur réseau peut retirer d'une bonne planification de ses réservations. Pour cela nous évaluons le coût de différentes stratégies, correspondant à la combinaison de différentes offres. Les résultats sont présentés Figure 1a. Lorsque l'opérateur ne considère que les offres R (stratégie R) cela signifie qu'il réserve pour toute la période assez de ressources pour faire face au pic de consommation le plus important, ici le 1^{er} janvier à minuit. Cela correspond à la stratégie traditionnelle des opérateurs, qui dimensionnent leur réseau pour faire face au pire cas. Puisque la plupart des ressources restent inutilisées la majorité du temps cette stratégie a un coût très élevé. Au contraire, si l'opérateur veut tirer partie au maximum de la flexibilité du cloud il peut s'affranchir de toute planification et acheter ses ressources lorsque le besoin se présente en ne prenant que des offres OD (stratégie OD). Le coût est bien moindre car chaque ressource payée est effectivement utilisée. Pour optimiser ses coûts l'opérateur peut prendre en compte plusieurs offres et utiliser notre modèle pour sélectionner la combinaison optimale. En utilisant à la fois des offres OD et R (stratégie OD+R) on observe un gain assez limité. En raison du faible trafic nocturne les ressources réservées ne sont pas utilisées tout le temps, ce qui conduit le modèle à en prendre assez peu, d'où une amélioration limitée par rapport à la stratégie OD. Si l'on considère en plus les offres S (stratégie R+OD+S), adaptées à la périodicité du trafic, le coût décroît fortement car le modèle en profite beaucoup. C'est le meilleur tarif que l'on puisse obtenir en ne considérant que les offres publiques retenues. Enfin, si l'on considère que l'opérateur utilise aussi son propre datacenter (stratégie R+OD+S+P), les coûts d'exploitation diminuent encore sensiblement. La répartition des réservations pour cette dernière stratégie est représentée Figure 1b. La nette diminution des coûts observée pour les stratégies OD+R+S et OD+R+S+P comparé aux stratégies R et OD démontre l'intérêt d'utiliser un modèle dédié plutôt qu'une stratégie traditionnelle inadaptée ou une absence de planification.

Cependant, la diminution du coût d'exploitation pour la stratégie R+OD+S+P masque les investissements de l'opérateur pour construire son datacenter. Notre modèle ne permet pas d'estimer si un tel investissement est rentable, cependant nous pouvons simuler l'utilisation instantanée du datacenter, afin de mesurer si ses capacités sont bien utilisées. La Figure 1c présente la consommation du CPU en fonction du temps. Cette utilisation n'est pas de 100%, car le trafic de nuit est très faible, mais elle reste élevée en semaine (70%-80%). Cependant, on constate une forte chute le week-end, particulièrement le dimanche, où le trafic est

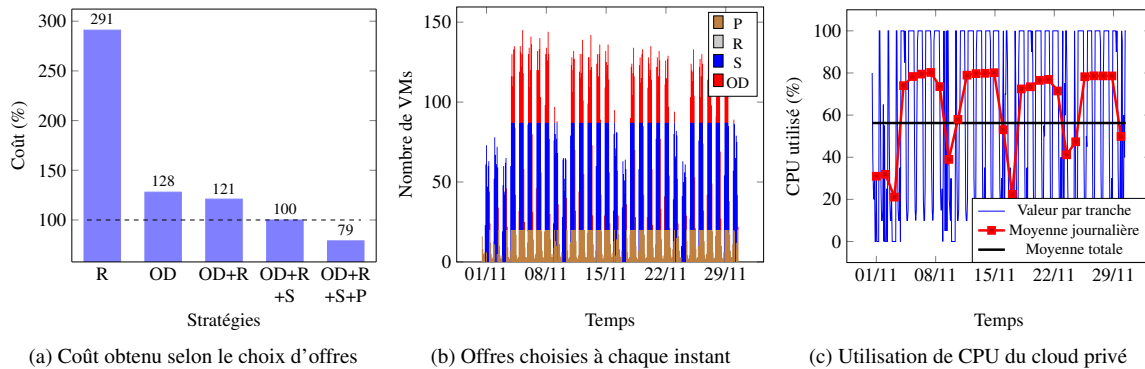


FIGURE 1: Résultat obtenu en appliquant notre modèle

moindre. Comme on le voit Figure 1b cela est dû au fait que les offres S, principalement réservées pour la semaine, mais toujours actives les week-end, sont suffisantes pour absorber la quasi totalité du trafic. Puisque ces offres sont réservées, les utiliser coûte moins cher que d'utiliser le datacenter privé. Ceci conduit à une moyenne d'utilisation plus basse que prévu sur la période : 56%. Ce résultat démontre l'importance pour un opérateur de prendre en compte tous les types d'offres publiques disponibles pour estimer l'opportunité d'investir dans des ressources propres.

Enfin, nous avons mesuré le temps d'exécution de notre modèle. Il croît linéairement avec la durée de la période considérée, la charge de trafic à gérer (*i.e.* le nombre de VNFCs à déployer par unité de temps) et la quantité d'offres de type OD à considérer. Il croît de manière exponentielle en fonction du nombre d'offres R et S disponibles, nous empêchant de considérer toutes les offres S possibles lors de nos tests et nous limitant à une option adaptée à la périodicité du trafic.

4 Perspectives

Nous avons proposé un programme linéaire aidant un opérateur réseau à sélectionner la meilleure combinaison d'offres (en termes de prix) permettant de réserver les VMs nécessaires pour supporter un ensemble de services réseau. Pour de futurs travaux nous envisageons tout d'abord le développement d'une heuristique afin d'accélérer le temps de résolution du problème, et en particulier pour les offres R et S. Cela permettrait de prendre en compte toutes les offres possibles sans préjuger de la périodicité du trafic, et d'améliorer encore les résultats. Ensuite, nous souhaitons achever le placement des VNFCs. Le modèle actuel indique simplement sur quel type de VM il faut placer le VNFC, ce qui offre un placement correct mais pas nécessairement optimal. Il faudrait développer un modèle dédié qui attribue à chacun une VM spécifique, et qui viserait par exemple à limiter le transfert d'un VNFC d'une VM à une autre.

Références

- [AKHA⁺17] Imad Alawe, Adlen Ksentini, Yassine Hadjadj-Aoul, Philippe Bertin, and Amelie Kerbellec. On evaluating different trends for virtualized and SDN-ready mobile network. In *2017 IEEE 6th International Conference on Cloud Networking (CloudNet)*, pages 1–6, Prague, Czech Republic, September 2017. IEEE.
- [BDNL⁺15] Gianni Barlacchi, Marco De Nadai, Roberto Larcher, Antonio Casella, Cristiana Chitic, Giovanni Torrisi, Fabrizio Antonelli, Alessandro Vespignani, Alex Pentland, and Bruno Lepri. A multi-source dataset of urban life in the city of Milan and the Province of Trentino. *Scientific Data*, 2(1):150055, December 2015.
- [ETS14] ETSI GS NFV-MAN 001. Network Functions Virtualisation (NFV); Management and Orchestration. 12 2014. Version 1.1.1.
- [MTC⁺20] Cedric Morin, Geraldine Texier, Christelle Caillouet, Gilles Desmangles, and Cao-Thanh Phan. Optimization of Network Services Embedding Costs over Public and Private Clouds. In *ICOIN 2020*, Barcelona, Spain, January 2020.