Inter-Datacenter Virtual Capacity Services: Reality and Mechanisms

P. Kokkinos, I. Gravalos, A. Kretsis
Computer Engineering and Informatics Department
University of Patras
Patras, Greece

E. Varvarigos
Electrical and Computer Engineering Department
National Technical University of Athens
Athens, Greece

Abstract—**The recent developments in network programmability and flexibility, shape the environment for the creation and offering of networking capacity services. Through such services one will be able to reserve on demand raw network capacity and for a specific duration, in the form of virtual links or of virtual networks that interconnect branch offices, datacenters or devices around the world. In this work, we initially present the capacity services landscape as it is formulated and discuss on the relations with the cloud computing service model. Next, we propose optimal Integer Linear Programming (ILP) based mechanisms for matching network demands to networking capacity services, in the same way users' computing requirements can be matched to virtual computing instances. We show through simulations that the proposed mechanisms can optimize the bandwidth usage, while minimizing the cost associated with the use of the network resources. We also investigate the effects of malleable capacity requests and of dynamic pricing.**

***Keywords:** inter-datacenter networking; capacity services; flexibility; pricing; cloud computing*

I. INTRODUCTION

Public cloud data centers are being built in various locations around the globe, providing computing and storage resources. Many of these datacenters are quite large in size, namely hyperscale. [1] identified 24 hyperscale operators that have in total 297 data centers all over the world, while it is expected that these will account for 83 percent of the public cloud server installed base in 2020.

Cloud applications and related operations like mirroring, disaster recovery and geographic redundancy [2] make necessary the efficient interconnection of the respective datacenters, supporting their throughput and availability requirements. Actually, traffic between data centers will account for almost 9 percent of total data center traffic, up from 7 percent at the end of 2015 [1], while it is growing faster than either traffic to end users or traffic within the data center

The inter-datacenter networking performance is also a critical criterion in selecting a cloud/network provider [3]. Prices vary widely by region due to differences stemming from available supply, competition, and cost of incremental upgrades. In Q4 2015, the median 10 Gbps price between Los Angeles and Sydney was 4.3 times that of the Los Angeles-Tokyo route, while the Miami-São Paulo route had 5.4 times the price of a wavelength between London and New York [4]. Another important characteristic is the network speeds for upload and download from a datacenter to the regional users and the related latency. According to [1] Asia Pacific leads all regions with an average fixed download speed of 33.9 Mbps and average fixed network latency with 26 ms, followed by Central and Eastern Europe with 30 ms.

Hyperscale datacenters are connected using high speed optical connections both landline and subsea. These are supported by advances in networking and management technologies including flexible optical networks [5] and Software Defined Networking - SDN [6]. Flexi-grid optical networks and SDN provide the ability to a network operator to decide on the exact network characteristics of its network, dynamically and from a central point [7].

This leads to the creation of the capacity services arena, through which it will be possible to reserve on demand and use in a number of minutes or less, raw network capacity and for a specific duration, in the form of virtual links or virtual networks, interconnecting virtual computing instances or real devices, all around the world. Actually, today we have already started viewing some proof of this not so far apart future. Pacnet Enabled Network in the Asia-Pacific region, provides scalable bandwidth and software-enabled intelligence, allowing customers to dynamically provision bandwidth (in 25 Gb/s, 37.5Gb/s, 50 Gb/s or 100 Gb/s increments) in minutes through a custom portal based on their business needs, with deactivation at the end of a customer-specified time period [8][9].

In this work, we leverage on the network capacity services notion and propose optimal Integer Linear Programming (ILP) based mechanisms for matching network requirements to the offered capacity services, similarly to the cloud computing model. The goal is to minimize the total cost of use and maximize the utilization of the interconnected reserved network. We assume a multitude of network service providers offering a diversity of virtual links between any pair of nodes. A number of quantitative and qualitative attributes differentiate those virtual links, necessitate the deployment of a virtual link decision method, with the aim of network resources' optimization. The performed simulation results indicate the importance of global network optimization in favor of the users and of the network providers, considering the fact that the international resources (e.g., submarine optical networks) are scarce and costly to build and maintain.

The reminder of this work is organized as follows. In Section II we describe the envisage networking services model and

978-1-5090-4026-1/17/$31.00 ©2017 IEEE

present an ILP formulation for matching network requests to offered capacity services. Simulation results are presented in Section III. Finally, in Section IV we conclude our work.

II. CAPACITY SERVICES SELECTION

A. *Virtual capacity services*

The ability of the network to provide on demand dedicated capacity is expressed under various names such as bandwidth calendaring [10], network slicing [11], network virtualization [12], virtual network embedding [13] and other.

In this work, we take an abstract view of the available network resources, following the cloud computing model in which users *select* among a predefined set of virtual computing instances [14] , with varying characteristics (CPU, memory, price, region etc) and without considering how these offerings are actually implemented in reality. In the same way, we assume that virtual capacity services are provided, characterized by the network capacity [9], the locations that they interconnect, their price and other parameters (modeled in what follows). A user *selects* a particular virtual capacity service based on the needs and creates a respective virtual capacity service instance.

We also expect that multiple global providers will exist (check also Section II), which will provide different offerings for virtual capacity services connecting the same locations: price, capacity granularities, reliability, access latency and bandwidth and other. A user should be able to indistinguishable utilize all the available services. This is similar to the utilization of multiple cloud providers, in the form of federated clouds [15].

The duration of using a virtual capacity service is also important. Cloud computing, follow the pay as you go model, where users utilize computing instances for as long as they need them and pay accordingly. This model can also be used in the network capacity services arena. However, considering also the scarcity of some network resources (e.g., such as subsea links) the duration of the usage may be limited or part of the virtual capacity service *selection* process, providing a specific set of reservation durations to select from. In this way the network operator will be able to increase the predictability of its network utilization and performance.

Overall, the envisaged networking services will enable small or medium companies that could not afford of signing multi-year contracts with network operators for some amount of fixed bandwidth, to now buy on demand dedicated network capacity in affordable and flexible prices, and to transfer their data efficiently all around the world. Such services will also lower total cost of ownership for network operators and enable them to offer novel networking products and entirely new pricing regimes, similar to those of the cloud providers.

B. *Modeling*

In our work, we consider E world regions were datacenters are located. Assuming that there are (s, d), s, $d \in E$ pairs, with data exchange requirements, we define a respective set $CP = \{w_1, \cdots, w_N\}$ in which every w, namely route, represents a communication pair (CP), not necessarily directly connected. In practice not all regions are directly connected, so only some of the possible routes are available, providing dedicated capacity services. For each $w \in CP$ route multiple virtual links $L_w = \{lw_1, \cdots, lw_{Mw}\}$ exist, e.g., from different providers. Each such virtual link l_w can be a single or multiple optical paths (lightpaths), protected or unprotected, utilizing subsea infrastructures either totally or partially. In our analysis, we assume that these characteristics are only reflected in the cost of using the respective virtual links.

We also assume that a variety of Virtual Capacity Services (CS) can be provided, each described by the following characteristics

$$CS_i = \{w, l_w, B, M, AL, AB\}, i \in [1, \cdots, K]$$

, where K is the number of such services offered. Each CS is characterized by the route w, the respective virtual link l_w where it operates and its capacity B, measured in Gbps. The provided capacities B are selected among a predefined set $\{B_1, \cdots, B_V\}$, where a higher number of available capacities V, indicates capacities of larger granularity. The definition of this set is actually a provider's concern and depends on the employed (e.g., optical) technologies, the operators pricing policies and the network performance objectives (such as in relation to the fragmentation and the utilization of the available spectrum).

Besides communication links, the landing datacenters' operational characteristics can also be taken into account. Thus, we also consider the access latency $AL = \{al_1, \cdots, al_R\}$ and the associated bandwidth $AB = \{ab_1, \cdots, ab_R\}$ provided by the landing datacenter (e.g., hyperscale datacenter) of a virtual link l_w, to the users/clients/other datacenters of the corresponding landing region (R). The cost M of a CS, is measured in cost units per hour. M can be a single value, depending on the provided B, or a more complex function that depends on the particular l_w, the provided AL, AB and other parameters.

C. *ILP Formulation*

We formulate the capacity services selection problem considering the scenario of a company interested in setting up a virtual network consisted of independent virtual links. This virtual network will connect the company's branch offices around the world or the datacenters where data are stored and services are provided to the regional users. This can be described by a set of requests:

$$S = \{R_1, \cdots, R_H\}$$

$$R_j = \{w, RB\}, j \in [1, \cdots, H]$$

, where each request includes the route w and the requested bandwidth RB.

Table I presents the proposed ILP formulation that matches user requests to the provided virtual capacity services, while minimizing the associated cost. Let us here also mention that the requested bandwidth may be not exactly match the capacity offered by a *CS*. Since a virtual link cannot be split into smaller portions, occupying a CS may result into respective bandwidth overprovisioning. Hence, our aim will also be to optimally allocate the available CSs in order to efficiently utilize the available resources. We include in the following formulation two variations of the problem: in the first each bandwidth request sets a hard constrain [Eq. (3)], while in the second is a malleable one [Eq. (4)].

Table I - Capacity services selection ILP formulation

ILP formulation

Input

E : Regions were datacenters are located
N: Number of routes between regions
K: Number of provided Virtual Capacity Services
H: Number of different requests

CS: The set of different types of Capacity Services $CS = \{CS_1, \cdots, CS_K\}$
$CS_i = \{w_i, l_w, B, M\}, i \in [1, \cdots, K]$

S : The set of requested services $S = \{R_1, \cdots, R_H\}$
$R_j = \{w_j, RB_j\}, j \in [1, \cdots, H]$

$B_i, i \in [1, \cdots, K]$: Capacity of virtual capacity service of type *i*
$M_i, i \in [1, \cdots, K]$: Cost of virtual capacity service of type *i*
RB_j: Requested capacity of request *j*
W,Q: weighting coefficients for the cost functions

Variables:

$X_{j,i}$: Boolean variable equal to 1 if *j*-th request is served with *CS* of type *i*, equal 0 otherwise
$Ts_{j,i}$: Integer variable equal to the difference between the requested bandwidth RB_j and the provided capacity B_i

P: The total network capacity used for serving all requests *S*

$$P = \sum_{j=1}^{H} \sum_{i=1}^{K} X_{i,j} \cdot B_i$$

C: The total cost of utilizing the *H* in number *CSs* and serving all requests *S*

$$C = \sum_{j=1}^{H} \sum_{i=1}^{K} X_{i,j} \cdot M_i$$

DU : Desired/targeted utilization for serving all requests *S*

$$DU = \sum_{j}^{H} RB_j$$

ILP formulation

Minimize

$$W \cdot C + (1 - W) \cdot (P - DU) \quad (1)$$

or

$$Q \cdot C + \sum Ts \quad (2)$$

Constraints

1. New instance assignment

$$\sum_{i=1}^{K} X_{i,j} = 1 \quad \text{for each } j \in [1, \cdots, H]$$

2. Instance capacity constraint

$$X_{i,j} \cdot RB_j \leq B_j \text{ for all } \quad i \in [1, \cdots, H] \text{ and } \quad j \in [1, \cdots, K] \quad (3)$$

Or

$$RB_j - B_i \leq B_i \text{ for all } \quad i \in [1, \cdots, K] \text{and } j \in [1, \cdots, H]$$

$$-(RB_j - B_i) \leq B_i \text{ for all } \quad i \in [1, \cdots, K] \text{and } j \in [1, \cdots, H]$$

(4)

3. Select links of the same source-destination pair as the request

$$X_{i,j} \cdot w_j = w_i \text{ for all } \quad i \in [1, \cdots, H] \text{and } j \in [1, \cdots, K]$$

The first constraint ensures that each request will be served by a single virtual service. The second constraint has actually two variations based on the instance of the problem (along with the respective objective) that we consider. In the first case the constraint guarantees that the selected virtual link's capacity will be sufficient for the serving the demand. In the second case, the constraint ensures that the difference between the provided and the requested capacity will be bounded by T_s. In this way we can introduce some flexibility by selecting virtual links of lower that the request capacity but also of lower cost, if of course the problem description allows it. The third constraint provides that each request will be served by a virtual link of the respective route.

There are also two variations of the ILP's objective. In the first case the objective is to minimize the reserved capacity and the associated cost. The *W* parameter can be used to tweak the importance given to these parameters. *W* close to 1 makes the cost for using the virtual links the dominant optimization parameter, leading to the selection of virtual links with low cost, neglecting the resource utilization criterion. *W* close to 0 makes resource utilization the dominant optimization parameter. In the second case the objective is to minimize the cost of using virtual links along with the total amount of excess or shortage of requested bandwidth in relation to the bandwidth provided by the selected virtual links. Again the *Q* parameter is used to tweak the objective's parameters. The number of variables and constraints in the above ILP formulation depends on the number *K* of offered virtual services and the number *H* of requests.

Also, the above ILP formulation can be extended, including the *AL* and the *AB* characteristics in each link and trying to minimize the access latency and bandwidth of the virtual link selected to serve a particular request. Time is another very important parameter that may affect the start time and the duration of the requests. However, in this work we assume that all requests ask service on a particular - the same - time period.

III. RESULTS

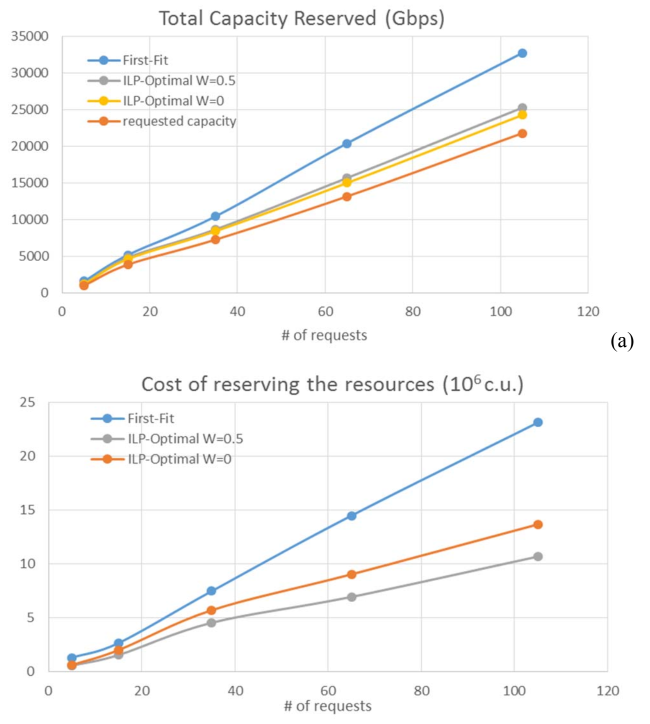
The proposed ILP based mechanism, was implemented in python using the PuLP library [16] and evaluated against various parameters. The main input parameters include: the number of regions (where a datacenter or presence of point exists), the regions' pairs with a connection (that is the routes), the min/max cost of the capacity services, the min/max capacity of the virtual links, the min/max requested capacity, the number of capacity service requests (for all routes).

In what follows, we name ILP-Optimal the variation of the ILP mechanism that uses objective (1) and constraint (3) and ILP-Optimal Flexible the variation using objective (2) and constraint (4). Also, for comparison purposes, we implemented in python a simple algorithm called "First-Fit", where each request is served from any virtual capacity service with sufficient capacity.

A. Basic performance

Figure 1.a shows the total capacity reserved for various number of requests *H*, comparing the ILP-Optimal and the First-Fit mechanisms, while also showing the total capacity requested.

The ILP-Optimal mechanism reserves capacity close to the requested one, and smaller than the First-Fit mechanism, avoiding overprovisioning of resources. Generally, more capacity is reserved than that actually requested due to the virtual links' capacity granularity that cannot match exactly to the requested capacity. The capacity reserved increases as expected with the number of requests. Figure 1.b shows the total costs (measured in millions cost units – c.u.) of reserving the respective resources where again the ILP-Optimal outperforms the First-Fit algorithm. We also performed experiments with various values for the *W* weight parameter. When *W*=0 the ILP-optimal tries to minimize only the reserved capacity, while when *W*=0.5 both cost and capacity utilization are minimized. We observe that in the particular simulation settings the difference, for *W*=0 and *W*=0.5, in the capacity reserved is small, while there is a more distinct variation in the total cost for serving the respective requests.



(a)

(b)

Fig. 1. (a) The total capacity reserved (measured in Gbps) and (b) the cost for serving the requests (measured in 10^6 cost units).

B. *Capacity services granularity*

In Figure 2 we performed experiments considering various granularities of the offered capacities. In particular, we change the number of offered capacities from 2 to 12, uniformly distributed in the range [20,400] Gbps. We observe that the total capacity reserved, both for the ILP-Optimal and First-Fit mechanisms, reduces as the granularity increases, indicating that more efficient capacity reservations are performed for the same set of requests, without the need for overprovisioning. After a point both algorithms reach an optimum reservation value, and any increase of the granularity does not affect further their performance. In all cases the ILP-Optimal outperforms the First-Fit algorithm. In any case, though larger capacity granularities increase reservation efficiency, they will also increase the mechanisms' execution times and the cost for the provider since these bandwidth granularities need to be matched with the respective available transponder capabilities. We should also note that in each iteration of the simulation, the input parameters are the same, except from the capacity services granularity and as a result the total requested capacity remains constant.

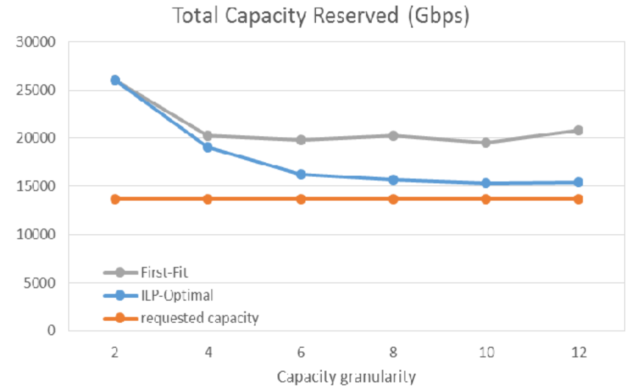
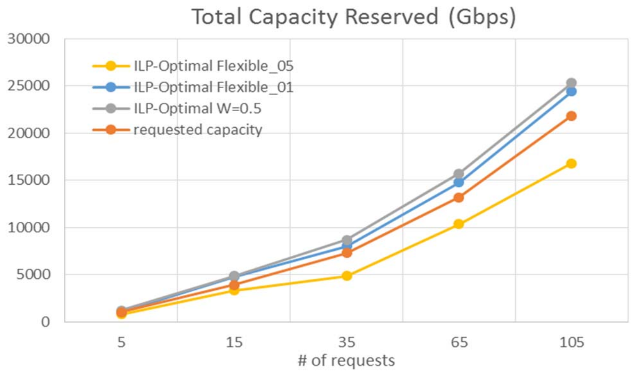


Fig. 2. The total capacity reserved (measure in Gbps) in terms of the provided capacity services' granularity.

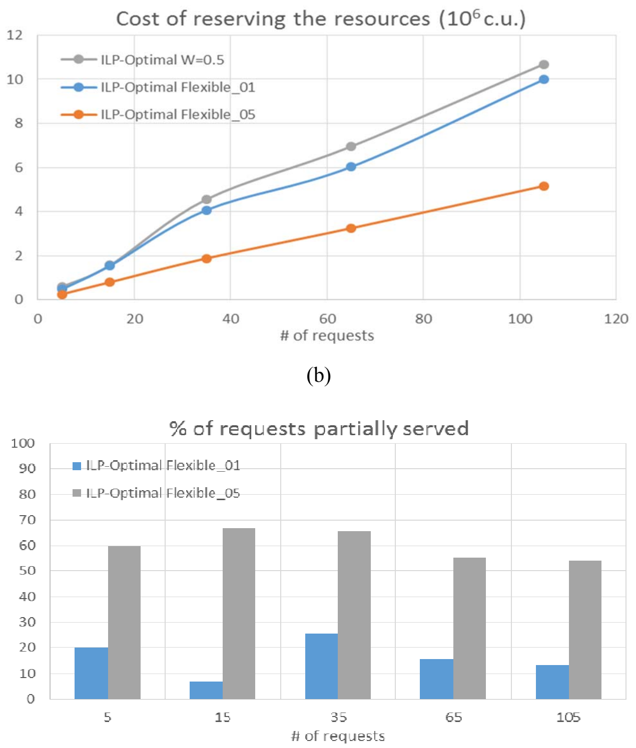
C. *Flexible bandwidth requests*

In the above we assumed that in all cases the full requested capacity is provided [based on the ILP constraint (3)]. In what follows, we evaluate the ILP-Optimal Flexible mechanism, in which the requests can be partially served, in terms of the reserved capacity [based on ILP constraint (2)] and using objective (4). We also use a flexibility parameter (Q = 0.001 and 0.005) that defines the extent to which the requests can be partially served.

Figure 3.a and Figure 3.b present the total capacity reserved and the respective cost, using the ILP-Optimal (W=0.5), ILP-Optimal Flexible (0.001 and 0.005) mechanisms. We observe that the flexible mechanisms (Flexible_01 and Flexible_05 respectively) can reserve bandwidth close to the requested, with however large economic benefits. Figure 3.c shows the number of requests not being completely served, in terms of the requested bandwidth.



(a)



(b)

(c)

Fig. 3. (a) The total capacity reserved (measure in Gbps) and (b) the cost for serving the requests (measured in 10^6 cost units), (c) the % of requests partially served, in terms of the reserced capacity.

D. *Dynamic network pricing*

Dynamic network pricing, it will also be part of the capacity services as a way to make their use more widespread and attractive. Time is of course a way to apply dynamic pricing by providing cheaper network prices in time periods were the traffic is generally low, enforcing in this way a kind of network balancing triggered by network prices. Figure 4 shows the total cost required for serving the demands, assuming either static network pricing (# periods = 1) or dynamic with different network prices per period (# periods 3 or 5).

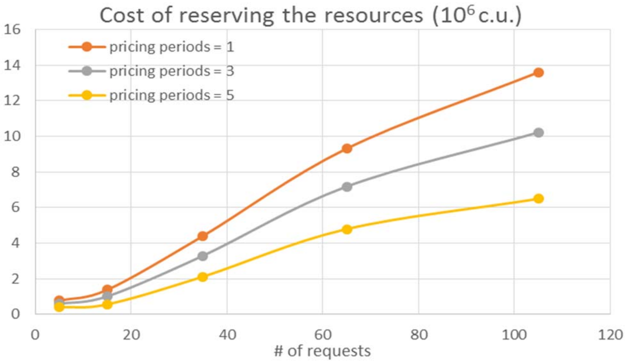


Fig. 4. The total cost for serving the requests (measured in 10^6 cost units), assuming dynamic network pricing.

IV. CONLCUSIONS

Today, the on demand provisioning of network capacity becomes a reality. The resulting capacity services will follow the cloud computing model, where users pay only for the capacity they need and for the time they use it. We formulate using an Integer Linear Programming (ILP) model, the problem of matching network demands to capacity services offered by multiple providers and propose respective mechanisms. We expect that such selection mechanisms will be mandatory for the operation of future capacity services. We show through simulations that the proposed mechanisms minimize the cost of using the resources, while maximizing their utilization and providing flexibility. We also exhibit the benefits of dynamic network pricing.

ACKNOWLEDGMENT

This work was partially supported by the EC through the Horizon 2020 Nephele project (g.a. 645212).

REFERENCES

[1] "Hyperscale Data Center Market - Global Opportunity Analysis and Industry Forecast, 2014 - 2022", Market Research Reports, Inc, 2016.

[2] P. Kokkinos, D. Kalogeras, A. Levin, E. Varvarigos, “Survey: Live Migration and Disaster Recovery over Long-Distance Networks”, ACM Computing Surveys, vol. 49, no. 2, 2016.

[3] Serverdensity, Network performance at AWS, Google, Rackspace and Softlayer, blog.serverdensity.com/network-performance-aws-google-rackspace-softlayer, Retrieved February 2017.

[4] TeleGeography, www.telegeography.com, Retrieved February 2017.

[5] K. Christodoulopoulos, I. Tomkos, E. Varvarigos, "Elastic Bandwidth Allocation in Flexible OFDM-based Optical Networks", IEEE/OSA Journal of Lightwave Technology, 2011.

[6] Open Networking Foundation, https://www.opennetworking.org, Retrieved February 2017.

[7] M. Channegowda, R. Nejabati, and D. Simeonidou, "Software-Defined Optical Networks Technology and Infrastructure: Enabling Software-Defined Optical Network Operations", Journal of Optical Communications and Networking, vol. 5, pp. A274-A282, 2013.

[8] Pacnet – network virtualization, https://www.infinera.com/pacnet-launches-first-network-virtualization-for-the-optical-layer-with-infinera-open-transport-switch/, Retrieved February 2017.

[9] Infinera Instant Bandwidth, www.infinera.com/technology/instant-bandwidth, Retrieved February 2017.

[10] L. Gkatzikis, S. Paris, I. Steiakogiannakis and S. Chouvardas, "Bandwidth calendaring: Dynamic services scheduling over Software Defined Networks", IEEE ICC, pp. 1-7, 2016.

[11] K. Samdanis, X. Costa-Perez and V. Sciancalepore, "From network sharing to multi-tenancy: The 5G network slice broker", IEEE Communications Magazine, vol. 54, no. 7, pp. 32-39, July 2016.

[12] Q. Duan, Y. Yan and A. V. Vasilakos, "A Survey on Service-Oriented Network Virtualization Toward Convergence of Networking and Cloud Computing", IEEE Transactions on Network and Service Management, vol. 9, no. 4, pp. 373-392, December 2012.

[13] A. Fischer, et al., "Virtual Network Embedding: A Survey", IEEE Comm. Surveys & Tutorials, vol. 15, no. 4, pp. 1888-1906, 2013.

[14] Amazon EC2 pricing, aws.amazon.com/ec2/pricing/on-demand/, Retrieved April 2017.

[15] B. Rochwerger et al., "The Reservoir model and architecture for open federated cloud computing", IBM Journal of Research and Development, vol. 53, no. 4, pp. 1-11, 2009.

[16] Optimization with Pulp, https://pythonhosted.org/PuLP/, Retrieved April 2017.