Cost Analysis of Initial Deployment Strategies for Virtualized Mobile Core Network Functions

The authors analyze the cost incurred by two "constraint-based heuristically applied" initial VNF/VNFC deployment strategies with reference to a virtualized mobile network infrastructure providing EPCaaS (evolved packet core as a service) while taking into consideration functional and administrative constraints. The cost of deployment is measured in terms of the utilization of data center infrastructure resources such as compute and networking.

Faqir Zarrar Yousaf, Paulo Loureiro, Frank Zdarsky, Tarik Taleb, and Marco Liebsch

Abstract

A virtualized network infrastructure is composed of multiple virtualized network functions (VNF) interconnected by well defined interfaces, and thus forming a VNF-graph. The initial deployment of such a VNF-graph inside a data center (DC) is a complex task with multidimensional aspects compared to deploying a single VNF that may represent a single network function. The problem space becomes more complex when each VNF is further decomposed into multiple VNF components (VNFC), where each VNFC embodies a subset of network functions attributed to the respective VNF. The challenge is to ensure that the deployment strategy meets the intra-functional constraints between the multiple VNFCs constituting the VNF-graph while ensuring service, performance and operational integrity, and also ensures optimal utilization of the underlying resources of the DC infrastructure (DCI).

In this article we analyze the cost incurred by two "constraint-based heuristically applied" initial VNF/VNFC deployment strategies with reference to a virtualized mobile network infrastructure providing EPCaaS (evolved packet core as a service) while taking into consideration functional and administrative constraints. The cost of deployment is measured in terms of the utilization of DC infrastructure resources such as compute and networking. We also present the discussion in view of the ETSI NFV MANO framework, undergoing standardization, that is responsible for management and orchestration of NFV systems including VNF deployment.

Faqir Zarrar Yousaf, Paulo Loureiro, and Marco Liebsch are with NEC Laboratories Europe.

Frank Zdarsky is with Red Hat GmbH.

Tarik Taleb is with Aalto University.

Introduction

Background

Network function virtualization (NFV) is fast emerging as a promising technology that leverages the concept of cloud technology and virtualization techniques into the realm of telecommunication networks. Mobile network operators are especially interested in exploring the potential of adopting this technology to enhance their competitiveness and reduce capital and operational costs. At present the network function entities are developed and ported on customized hardware platforms designed and tested for meeting the functional and operational requirements for a specific function or set of functions. Such a rigid infrastructure makes network scalability difficult and expensive and increases the total cost of ownership (TCO). Additionally it also locks the operator into specific hardware and/or software vendors, while constraining the operator from rolling out new services as per demand, thereby impacting revenue.

NFV technology has the potential of offsetting the above issues while providing a highly scalable, flexible, and elastic network infrastructure. NFV involves the virtualization of network node functions and hosting these virtualized network functions (VNF) on virtual machines (VM), which in turn are deployed on commodity servers (i.e. COTS servers). These VNFs are then interconnected across the servers to provide the intended network services (NS). For example a mobile core network, such as evolved packet core (EPC), is composed of several functional entities interconnected via standardized interfaces. In order to virtualize an EPC for providing EPC as a service (EPCaaS), the functional elements are characterized by multiple VNFs, where the respective VNFs are interconnected via well-defined interfaces forming a VNF-graph. For truly resilient and elastic performance, the VNFs can also be decomposed into multiple inter-connected VNF components (VNFC), where each VNFC instance is hosted on a single VM.

COMMUNICATIONS STANDARDS

ETSI NFV MANO System

The inherent advantages offered by NFV introduces the challenge of the management and orchestration of the multitude of distributed VNF(C)s deployed across multiple servers in a network functions virtualized infrastructure (NFVI), for example a data center (DC), to provide carrier-grade service. To this effect an ETSI ISG has been formed to standardize the various aspects of NFV enabled networks, including the NFV management and orchestration (MANO) framework [1]. The proposed MANO framework architecture is shown in Fig. 1, which is composed mainly of three functional blocks, namely the virtualized infrastructure manager (VIM), the VNF manager (VNFM), and the NFV orchestrator (NFVO), interconnected over specific reference points. There are additional data repositories that may contain necessary information about NS, VNF, NFV, and NFVI that will enable the NFVO to perform its tasks. The MANO architecture also defines reference points for interfacing the MANO system with external entities like NFVI, OSS/BSS, VNFs, and element managers (EM) for delivering unified management and orchestration of a VNF system.

An interfaces and architecture (IFA) WG has been formed under the ETSI NFV that has the mandate to develop specifications for the MANO framework. In this respect, the IFA WG at present is in the process of specifying interfaces, requirements, and operations for the reference points in view of the functional/operational scope

of the NFVO, NFVI, and VIM, as described in [1]. Besides traditional FCAPS management, the MANO framework focuses on newer management aspects introduced by NFV, such as the creation and life-cycle management (LCM) of the virtualized resources for the VNF, and collectively referred to as VNF management [1]. There are several VNF management tasks such as VNF scaling, migrating, and updating, to name a few, but the deployment/instantiation of VNF(C)s in a DC (i.e. in the NFVI) is the main focus of this article.

It is a challenging task to initially deploy VNF(C)s on a NFVI, owing to the intra-functional dependencies and constraints among the various VNF(C)s. Thus, during deployment the MANO system must take into consideration the intricate (anti)affinity between the various VNF(C)s that constitute a complex NS such as a virtualized evolved packet core (vEPC). Following are the main aspects that the MANO system must take into consideration when forming a VNF-graph and making deployment decisions when realizing complex NS such as the vEPC:

- Networks by themselves offer a complex, well connected, and well defined ecosystems, composed of multiple complex, yet well defined and well specified functions and with strict relationships.
- The network functions are interconnected to each other via well-defined interfaces and communicate with each other using well-defined and specified protocols.
- These network functions work in a coordinated manner to ensure end-to-end service integrity and connectivity.
- Each network function has a different set of system and resource requirements.
- Each network function has a well-defined functional scope of operation as stipulated by the relevant standards.
- Achieving carrier-grade performance from the deployed VNF-graph is still the number one priority for many mobile operators.

In this article we address and analyze the issue of initial deployment of a virtual mobile network platform, represented by a VNF-graph, within a DCI (i.e. NFVI). For our analysis we have adopted a simplistic architecture of a vEPC network [2] as a reference for our analysis. The main objective and motivation behind this article is to compare two constraint-based heuristic approaches of initial deployment of vEPC VNFCs over a DCI and analyze the impact of the two deployment strategies on the cost of deployment.

The rest of the article is organized as follows. The next section provides some related research work, by which we will provide a conceptual and functional overview of EPCaaS with reference to the vEPC network. This is followed by a description of the evaluation framework and method that includes the modeling of the DC and vEPC network. We describe the two proposed deployment strategies, and we present performance analysis. The article then concludes.

Related Work

Several pioneering research works have been conducted to enable the creation and runtime management of mobile networks over the cloud, studying different implementation options [3] and devising an entire framework for the creation of end-to-end mobile services, including mobile transport networks, on the cloud [4]. For a successful creation of mobile core networks on the cloud, algorithms for optimal placements of VNFs on a federated cloud and within the same DC are of crucial importance.

In traditional mobile core networks, mechanisms and algorithms have been devised to select, for mobile users, optimal data anchor gateways from within a fixed range of geographically static gateways for the sake of communication efficiency.

However, in cloud-based mobile core network, gateways are realized as VNFs, which are not only created on-demand, but operators have more flexibility in deciding where to place VNFs of gateways, rather than just selecting gateways from within a fixed set of static gateways. Such flexibility helps mobile operators to dynamically dimension, plan, and re-plan their mobile networks whenever there is a need for that and as per the changing behavior of mobile users, the features of the provisioned services, and according to other metrics relevant to the mobile network performance. Regarding the latter, the authors in [5] proposed a VNF placement method, particularly for creating mobile gateway functionalities (serving gateway (S-GW)) and their placement in federated clouds so that the frequency of S-GW relocation occurrences is minimized. In [5], the aim was to conduct an efficient planning of service areas (SAs) retrieving a trade-off between minimizing the user equipment (UE) handoff between SAs, and minimizing the number of created instances of the virtual S-GWs. In [6] the focus was on VNF placement and instantiation of another mobile network functionality, namely data anchoring or PDN-GW (P-GW) creation/selection. That work argued the need for adopting application type and service requirements as metrics for creating VNF instances of PDN-GW and selecting adequate virtual P-GWs for UEs receiving specific application

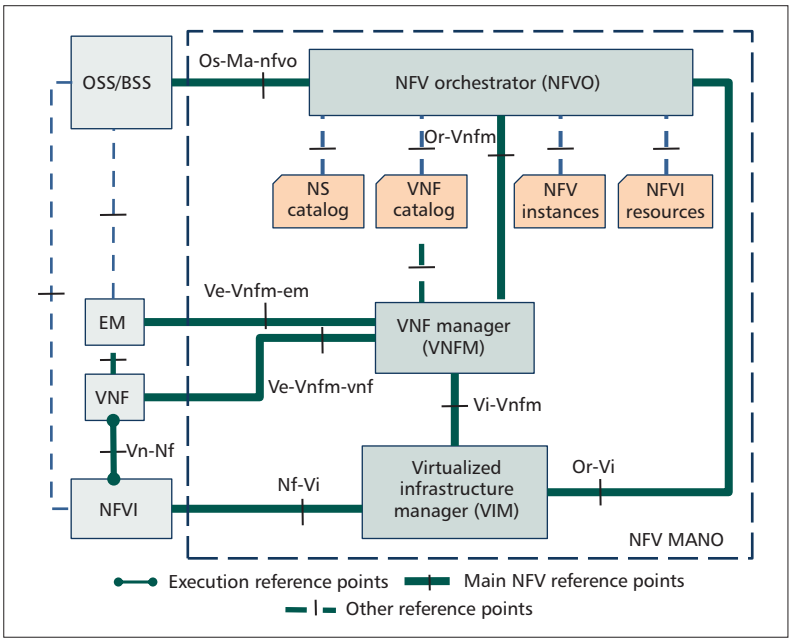

Figure 1. ETSI NFV management and orchestration (MANO) framework overview [1].

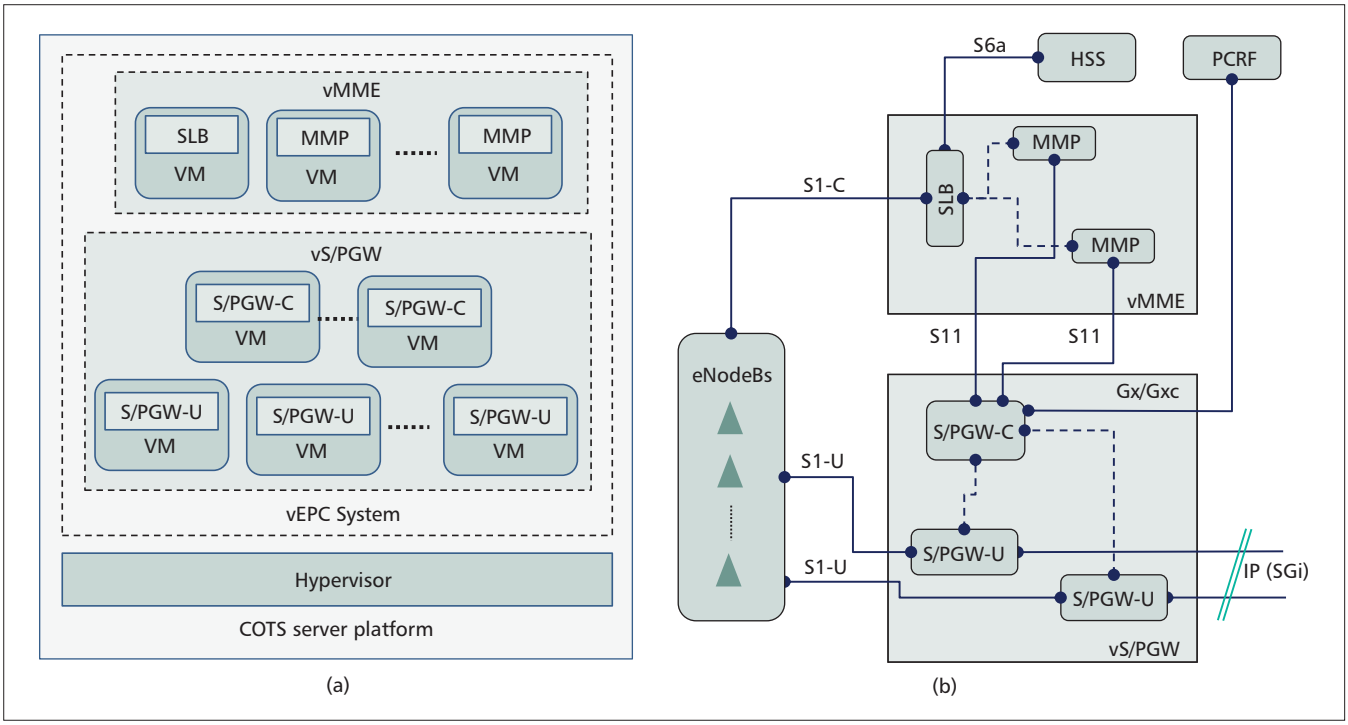

Figure 2. Virtualized evolved packet core (vEPC) system overview: a) functional overview; b) interfaces between vEPCs VNFCs.

types. The placement of P-GW VNFs was modeled through a nonlinear optimization problem whose solution is NP-hard. Three heuristics were then proposed to deal with this limitation. In [7] the authors proposed a softEPC framework for flexible and dynamic instantiation of EPC VNFs with reference to the actual traffic demand at appropriate topological locations.

While the above research works considered the problem of VNF placement across federated clouds, the present article will be looking into the VNF placement problem within the same DC. In this context research work has been conducted for decisions on VM placement within the same DC, having as the objective cost savings thanks to better utilization of computing resources and less frequent overload situations. In [8] performance isolation (e.g. CPU, memory, storage, and network bandwidth), resource contention properties (among VMs on the same physical host), and VMs' behavioral usage patterns are taken into account in decisions on VM placement, VM migration, and cloud resource allocations.

In other research works, optimal placement of VMs takes into consideration electricity-related costs as well as transient cooling effects [9]. Others do autonomic placement of VMs as per policies specified by the data center providers and/or users [10]. Other VM placement strategies consider maximizing the profit under a particular service level agreement and a predetermined power budget [11].

In [12] the authors take into consideration the reduction in control-signaling traffic and congestion in the data plane of a vEPC system. However, the authors in [12] take a different view, where instead of focusing on the placement of individual VMs, they propose to group multiple vEPC functions, for example SGW and PGW, in one VM, and then interconnecting the different VM segments via GTP to achieve the desired objective.

Thus instead of taking a one dimensional view of VM placement and focusing on the single optimization factor, the VNF placement problem, addressed in this article, is more complex. This is so because a virtualized network infrastructure is composed of multiple VNFs, which are interconnected by well-defined interfaces, thereby forming a VNF-graph. The problem space becomes more complex when a single VNF gets further decomposed into multiple interconnected VNFCs, and there is a strict functional relationship between the various VNFCs and performance constraints that makes the deployment process more complex.

Unfortunately, there is not much information available that may analyze the impact of deployment strategy during the initial deployment of a virtualized network infrastructure (represented by a VNF-graph) in a DC. In this regard, this article analyzes the impact on the cost of DC resources, such as networking and computing, by comparing the impact of two "constraint-based and heuristically-derived" deployment strategies, namely vertical serial deployment (VSD) and horizontal serial deployment (HSD) strategies adopted for the deployment of a virtualized mobile core network, referred to as a vEPC.

EPCaaS: Conceptual and Functional Overview

The objective of EPCaaS is to virtualize the EPC infrastructure in order to extend the advantages of the cloud system to mobile network operators.

This is done by instantiating the EPC system's functional entities (e.g. mobility management entity (MME), S-GW, P-GW) as VNFs on VMs over COTS servers instead of a specialized mission-specific, custom tuned, expensive hardware platform. To provide the service and design concept of EPCaaS we use a simplistic architecture of a vEPC network [2] as a reference use case. The overview of this architecture is depicted in Fig. 2a, where the MME and S/P-GW VNFs are referred to as vMME and vS/P-GW, respectively.

The following four possible architectural reference models for EPCaaS have been specified in [3, 13] based on how the VNFs are mapped on the VMs:

- 1:1 mapping, where each EPC VNF is implemented on a separate VM.
- 1:N mapping, where each EPC VNF is decomposed into sub-functional entities (i.e. VNFC) and each VNFC is implemented on a separate VM.
- N:1 mapping, where the complete EPC system is implemented on a single VM.
- N:2 mapping, which is similar to N:1 except that it separates the control plane (CP), user plane (UP), and database services of the EPC onto three separate interconnected VMs.

The vEPC system falls in the category of 1:N mapping, where the respective VNFs of the vMME and vS/PGW functions are decomposed into separate CP and UP VNFCs in order to render enhanced agility and elasticity in view of different traffic and application types. Thus the vS/PGW is divided into two VNFCs, namely vS/PGW-C and vS/PGW-U, with the former VNFC processing the CP load and the latter processing the UP load. Similarly, the vMME functionality is embedded in the combination of signaling load balancer (SLB) and mobility management processor (MMP) VNFCs, where the MMP performs the processing task of the MME. The combination of SLB and MMP will allow the scaling of vMME by using SLB and by adding/deleting MMPs. Each functional entity (i.e. SLB, MMP, vS/PGW-C, and vS/PGW-U) is realized on a separate VM, and the inter-connectivity between these VNFCs is based on standardized interfaces (Fig. 2b).

Evaluation Framework and Methodology

For cost analysis, we have developed an evaluation framework in C++. The evaluation framework has been designed with reference to the functional requirements of the MANO system [1]. This framework is composed of a DCI model, a vEPC system model, and a deployment model. For a specific CP/UP input load, the vEPC system model determines the required number of VNFCs and their respective resource requirements that will support the incident traffic load. The deployment model, based on a specific deployment strategy, will then deploy the respective vEPC's VNF-graph, including the respective VNFCs, on the servers of the underlying DCI model while taking into account the resource requirements of individual VNFCs and the vEPC system internal bounds and constraints. The framework then computes and determines the cost incurred by the respective deployment strategy in terms of DC networking and computes resource consumption for the incident CP/UP traffic load. Our evaluation framework can be scaled to any size DC and to any size vEPC system, depending on the load on the operator's network. The overview of the DCI model, the vEPC system model, and the deployment strategies, are discussed in the following sub-sections.

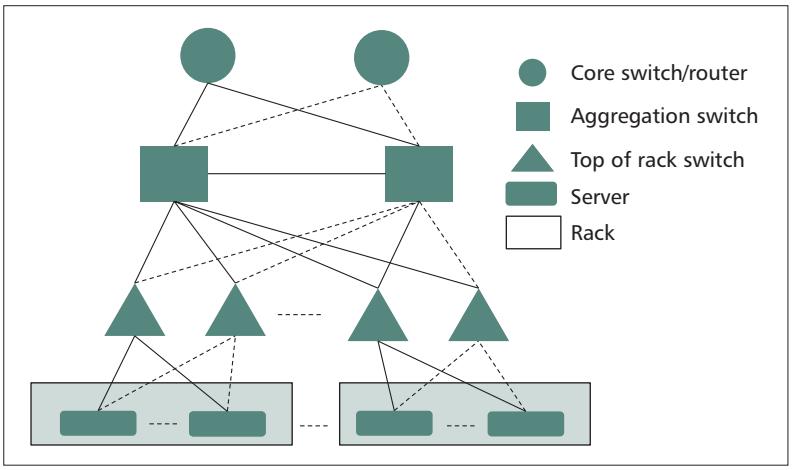

Figure 3. Three-layer date center infrastructure model.

Data-Center Infrastructure Model

For the analysis, we have modeled the traditional hierarchical three-tier DC architecture composed of:

- The core layer
- The aggregation layer
- The access layer [14]

At the lowest level is the access layer, which contains pools of servers housed in racks, where each server is connected to one (or two for redundancy) top-of-rack (TOR) L2 switch. Each TOR switch is, in turn, connected to one (or two for redundancy) high capacity L2/L3 switch at the aggregation layer. The aggregation switches are then connected to the top-level switches/routers, forming the core layer. Such a fat-tree topology can be scaled up, in turn, by scaling up each individual switch. Figure 3 illustrates the DCI topology that we have modeled, where the dotted lines indicate redundant links, thereby connected to the redundant/backup node. For our analysis, we do not consider failure scenarios and hence the redundant links/nodes are not utilized. The access layer is modeled as an $m \times n$ matrix where m is the number of racks and n is the number of servers per rack. For our analysis, we consider a homogenous access system where all racks are of the same size and all servers are of the same configuration and form-factor. The servers are modeled having x number of CPU cores and $xGbps$ aggregate network bandwidth. On the other hand, the switches/routers are modeled considering $xGbps$ aggregate bandwidths.

vEPC System Model

The vEPC system is modeled by characterizing the individual VNFCs (SLB, MMP, S/PGW-C, and S/PGW-U) in terms of the CP/UP load that they process. The model also captures the interfaces between the different relevant VNFCs, as depicted in Fig. 2b. Figure 2b illustrates the interconnected VNFCs constituting the vEPC system with relevant interfaces. The model is able to determine

Parameter	Notation	Value
Total CPU cores per VNFC	N_{core}^{VNFC}	4
eNB per SLB	$N_{eNB}^{SLB_i}$	100
Number of S/PGW-U per SPGW-C	$N_{SPGWU}^{SPGWC_i}$	6
Maximum S1C load per MMP	$L_{S1C,max}^{MMP_i}$	500,000 ev/hr
Maximum S11 load per SPGW-C	$L_{S11,max}^{SPGWC_i}$	1,000,000 ev/hr
Maximum S11 load per SPGW-U	$L_{S11,max}^{SPGWU_i}$	166666.7 ev/hr
CP load demand (ev/hr)	C_{cp}	$x * L_{S1C,max}^{MMP_i}$ where x = 0.25, 0.50, 0.75, 1.0
Number of eNBs, where each value corresponds to the respective value of C_{cp}	N_{eNB}	[1500, 2000, 2500, 3000]
Average CP packet size (in bytes)		192
Average number of messages per CP event		6
UP load demand (Gb/s)	C_{up}	64, 128, 256, 512
UP packet size (in bytes)		512

Table 1. Simulation parameters.

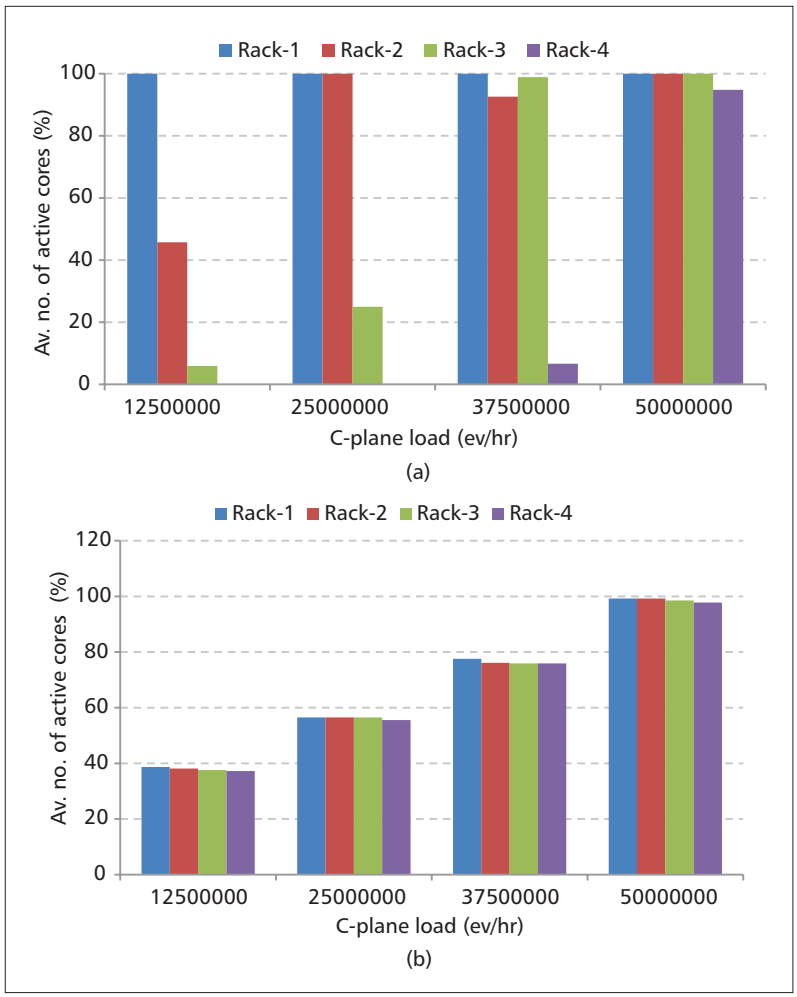

Figure 4. Average number of active cores per rack: a) VSD; b) HSD.

not only the number of relevant VNFCs required to handle a particular CP/UP load profile, but also determines the resource requirements of individual VNFCs in terms of CPU cores and network bandwidth. This information is then used to analyze the deployment cost of the vEPC system in a DCI, thereby enabling the operators to dimension the resources of their respective DCI for specific load conditions and service requirements.

This model is expected to provide insight into the resource requirement of every VNFC and thus the size of the overall vEPC system, in response to external inputs. The load models for vMME and vSPGW are derived with reference to Fig. 2b, and summarized below.

Load Model for vMME: As stated earlier, the vMME is composed of the SLB and MMP virtual instances. The load from the eNBs is balanced by SLB among multiple MMP instances. We assume SLB to be balancing the load between MMPs in an equally weighted round-robin manner. The total S1C load on a single SLB instance from eNBs is the aggregate S1C loads from all eNBs associated with the SLB.

Thus, the load on a single MMP instance is the total S1C load on a single SLB instance divided by the number of MMPs that a single SLB can serve. The ratio between the number of eNBs per SLB and the number of MMPs per SLB depends on the load balancing capability of the SLB as well as the maximum load that an MMP instance can handle.

Load Model for vS/PGW: The vS/PGW is modeled by characterizing the S/PGW-C and S/PGW-U VNFCs in terms of the CP and UP load that the respective VNFCs process. With reference to Fig. 2b, the total CP load incident on a single S/PGW-C instance is the sum of the CP loads from the policy and charging rules function (PCRF) and a proportion of the total S11 load from the associated MMPs.

Similarly, the total load processed by a single S/PGW-U instance is the proportion of the S11 load (i.e. CP load) from the S/PGW-C and the S1U load (i.e. UP load) from the associated eNBs (Fig. 2b).

Deployment Strategies

Following are the two constraint-based and heuristically derived deployment strategies:

- Vertical serial deployment (VSD) strategy
- Horizontal serial deployment (HSD) strategy

Both strategies deploy VNFCs serially such that the vMME VNFCs (i.e. SLBs and MMPs) are deployed first, followed by the vS/PGW VNFCs (i.e. S/PGW-C and S/PGW-U). In VSD, VNFCs are deployed from top to bottom on servers of one rack, and when no more resources are available in the rack, VNFCs are deployed on to the servers in the next rack. In HSD, VNFCs are deployed on the first available server in a rack, then moving on to the next available server on the next rack, and so on until all VNFCs are deployed. In other words, considering the access layer as an $m \times n$ matrix, in VSD, VNFCs are deployed column-wise, whereas in HSD the VNFCs are deployed row-wise.

While deploying, the VSD/HSD deployment strategy will take into account the (anti)affinity

between respective VNFCs, system reliability, server resources in terms of available CPU cores, and the network resources such as the capacity of the network interfaces on the servers and of the links in DCI. The following constraints are also taken into consideration during deployment:

- For reliability, a single server may not have more than one instance of the S/PGW-U belonging to the same logical vS/PGW.
- Each time a VNFC is instantiated, the associated standby VNFC will also be instantiated.
- A single server shall not host the active and standby instances of a particular VNFC.
- A VNFC is deployed only if the server has the CPU cores required by the target VNFC.

In both VSD and HSD, any server that may not have the resources required for a particular VNFC or does not offer affinity with any of the previously installed VNFCs is skipped over. For our analysis, and for the sake of simplicity, we assume that the servers are all dedicated for vEPC system deployment and no other third party services are running on them.

Performance Evaluation

In order to compare and analyze the cost impact of the VSD and HSD deployment strategies on the DCI computing and networking resources, we perform experiments on our evaluation framework using CP load (C_{cp}) and UP load (C_{up}) values based on conservative estimates during a busy hour period. According to [15], a MME can experience a sustained signaling load of 500 to 800 messages per UE during busy hours, and up to 1500 messages per UE per hour under adverse conditions. Furthermore, according to [16] the chattiest applications can generate up to 2400 signaling events per hour. Based on these observations, we assume 90 users per eNB that generate the bulk of traffic events. For our scenario, we assume 5 percent of users generating 2400 events/hr, 25 percent producing 800 events/hr, and 70 percent producing 500 events/hr during busy hours. Thus during busy hours a vEPC system will encounter 60300 events/hr from a single eNB. Based on the incident load, the vEPC system model with the help of equations 1-6 will compute the required number of VNFCs. These VNFCs are then deployed by the respective deployment strategy (i.e. HSD and VSD) on the DCI model in view of the constraints and affinity between the relevant VNFCs. The access layer of the DCI is modeled as a 4 × 45 matrix, and all of the 180 servers have 16 cores each.

For simplicity, we assume all VNFCs are assigned four CPU cores during deployment. Our evaluation framework also provides the standby VNFCs based on 1+N redundancy, but as we are not considering a failure scenario, we will not consider the standby VNFCs and corresponding links for throughput calculations. We also ignore the L_{PCRF} during calculations. The rest of the parameters used in our simulation framework are listed in Table 1, which are derived from equations 1-6 while based on assumptions described above.

The performance of two deployment strategies (i.e. VSD and HSD) are measured with respect to the average number of active cores utilized per rack (Fig. 4) and the average throughput per rack (Fig. 5) for four C_{cp} values.

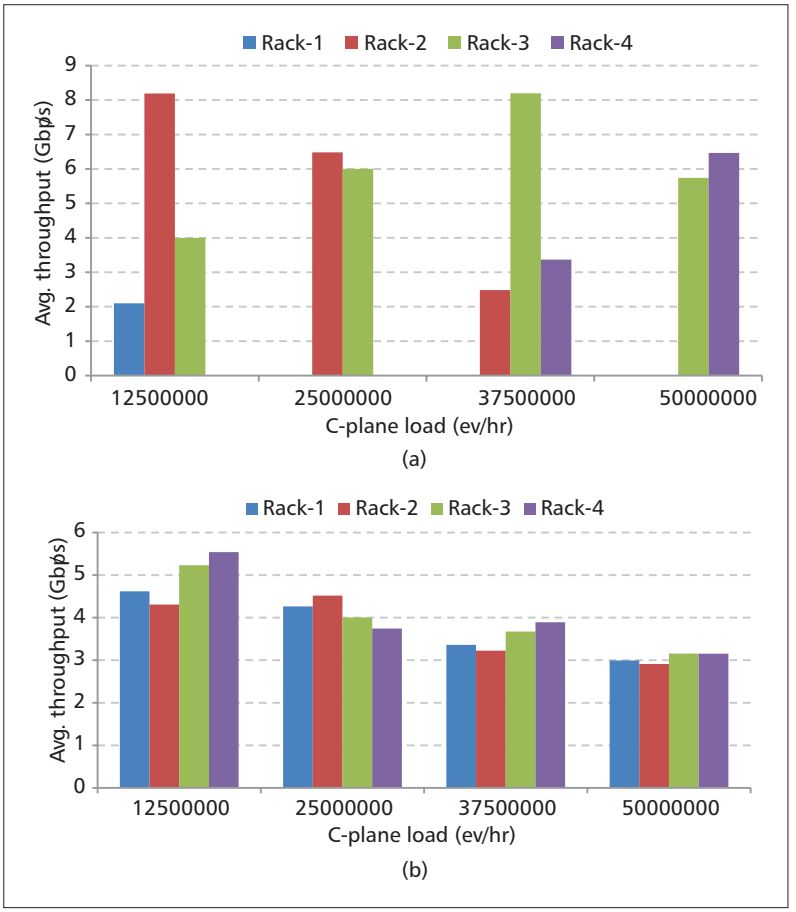

Figure 5. Average throughput per active server for UP load = 512 Gb/s: a) VSD; b) HSD.

As is evident, the deployment strategy has a marked and substantial effect on the distribution of the number of active cores on a per rack basis. With VSD (Fig. 4a), 100 percent of all cores (and hence all servers) are utilized in rack-1, while the cores in other racks become sequentially active with increasing load. As a result, there are load conditions where a rack (and hence the servers in it) may remain completely inactive and un-utilized. For example, the servers in rack-4 remain un-utilized for $C_{cp} = 12.5 \times 10^6$ *ev/hr* and $C_{cp} = 25 \times 10^6$ *ev/hr*. This will cause uneven load distribution over the access links and hence the ToR switches, where one link or switch may become overloaded, while the others may remain un/under-utilized. This is evident from Fig. 4a, where the load is unevenly distributed among servers in the four racks. For example, for $C_{cp} = 50 \times 10^6$ ev/hr, all the load is on servers on racks 3 and 4, whereas racks 1 and 2 have no load on them.

In contrast to VSD, HSD deploys VNFCs evenly across the racks, resulting in even and optimum utilization of the computing and networking resources under all load conditions. This can be observed from Fig. 4b, where under all load conditions, VNFCs are evenly deployed across the racks and thus the CPU core assignment is even. This will also ensure even distribution of load over the access links, and hence the ToR switches, as evident from Fig. 5b.

Thus ,in contrast to VSD, the HSD strategy results in the optimal utilization of the DCI

At present a scalable NFV deployment planning and auto-evaluation tool is being developed based on the work presented in this article. Such a tool is expected to aid the DC operators with a quick analysis of their deployment policy.

resources without overloading any particular set of servers, and the resources scale evenly with an increase in input load. This marked difference in performance is due to the fact that in HSD, VNFCs are deployed horizontally on available servers across different racks, as opposed to VSD, where all resources of one rack need to be allocated before moving to the next rack.

Conclusion

In this article we propose and analyze two deployment strategies, namely HSD and VSD, for initial deployment of multiple VNF(C)s constituting a VNI on the operator's DCI. The performance is analyzed in terms of the cost incurred by the respective deployment strategy, where the cost is measured in terms of the utilization of a DC's computing and networking resources. The analysis is presented with reference to deploying a vEPC NS for providing EPCaaS.

As is evident from the results, for specific load profile, the total number of active servers and active cores are the same for both HSD and VSD. However, HSD delivers the best performance in terms of even distribution of load over all servers, access links, and hence the ToR switches.

In fact, for higher load profiles, HSD will result in reduced average throughput per active server as the load is evenly distributed across all racks while the number of active servers increases. In contrast to HSD, VSD is not efficient as it causes uneven distribution of VNFCs and hence load on particular servers. This may make specific racks, and hence the servers therein and associated links, to be 100 percent utilized while some other racks with servers may remain underutilized, or not utilized at all.

At present a scalable NFV deployment planning and auto-evaluation tool is being developed based on the work presented in this article. Such a tool is expected to aid DC operators with a quick analysis of their deployment policy, thus enabling them to appropriately dimension their respective DCIs to meet the expected peak traffic demands while optimizing the utilization of available resources. In the near future this tool will be integrated as part of an NFV DevOps solution that is in the planning stage.

Acknowledgment

The research work presented in this article is conducted as part of the Mobile Cloud Networking project, funded by the European Union Seventh Framework Program under grant agreement no. [318109].

References

[1] ETSI NFV GS, "Network Function Virtualization (NFV) Management and Orchestration," NFV-MAN 001 v0.8.1, Nov 2014.
[2] White paper TE-524262, "NEC Virtualized Evolved Packet Core — vEPC; Design Concept and Benefits," 2015.
[3] T. Taleb *et al.*, "EASE: EPC as a Service to Ease Mobile Core Network Deployment over Cloud," *IEEE Network Mag.*, vol. 29, no. 2, Apr. 2015, pp. 78–88.
[4] T. Taleb, "Towards Carrier Cloud: Potential, Challenges, & Solutions," *IEEE Wireless Commun.*, vol. 21, no. 3, June 2014, pp. 80–91.
[5] T. Taleb and A. Ksentini, "Gateway Relocation Avoidance-Aware Network Function Placement in Carrier Cloud," *ACM Int'l. Conf. Modeling, Analysis & Simulation of Wireless and Mobile Systems (MSWIM'13)*, Nov. 2013.
[6] M. Bagaa, T. Taleb, and A. Ksentini, "Service-Aware Network Function Placement for Efficient Traffic Handling in Carrier Cloud," *IEEE Wireless Commun. and Net. Conf. (WCNC '14)*, Apr. 2014.
[7] F. Z. Yousaf *et al.*, "SoftEPC: A Dynamic Instantiation of Mobile Core Network Entities for Efficient Resource Utilization," *IEEE Int'l. Conf. Commun. (ICC'13)*, June 2013.
[8] G. Somani, P. Khandelwal, and K. Phatnani, "VUPIC Virtual Machine Usage Based Placement in IaaS Cloud," *Computing Research Repository (CoRR)*, vol. abs/1212.0085, Dec. 2012.
[9] K. Le *et al.*, "Reducing Electricity Cost Through Virtual Machine Placement in High Performance Computing Clouds," *ACM International Conference on High Performance Computing, Networking, Storage and Analysis (SC'11)*, article 22, 2011, 12 pages.
[10] C. Hyser *et al.*, "Autonomic Virtual Machine Placement in the Data Center," HP Labs technical report, 2007.
[11] W. Shi and B. Hong, "Towards Profitable Virtual Machine Placement in the Data Center," *4th IEEE Int'l. Conf. Utility and Cloud Computing (UCC'11)*, Dec. 2011.
[12] A. Shami ; M. Mirahmadi, and R. Asal, "NFV: State of the Art, Challenges, and Implementation in Next Generation Mobile Networks (vEPC)," *IEEE Network Mag.*, vol. 28 , no. 6, Dec. 2014, pp. 18–26.
[13] "D4.1: Mobile Network Cloud Component Design," *EU Mobile Cloud Networking Project*, Nov. 2013.
[14] Cisco, "Cisco Data Center Infrastructure 2.5 Design Guide," Nov. 2, 2011.
[15] D. Nowoswiat, "Managing LTE Core Network Signaling Traffic," Alcatel-Lucent, TechBlog, July 30, 2013.
[16] T. Parker, "Chatty Smartphones Stressing Networks," *Fierce Wireless Tech*, Apr. 27, 2012.

Biographies

Faqir Zarrar Yousaf (zarrar.yousaf@neclab.eu) is a senior researcher at NEC Laboratories Europe in Heidelberg, Germany. His current research interest includes NFV/SDN related technologies and its application to the emerging 5G network architecture. He has made several contributions to the ETSI NFV standardization body, and has more than 30 publications in international conferences/journals. He received a Ph.D. from TU Dortmund, Germany in 2010, and has two masters degrees from George Washington University, USA (2001) and the University of Engineering and Technology, Peshawar, Pakistan (1999).

Paulo Loureiro (paulo.loureiro@neclab.eu) is a senior researcher at NEC Laboratories in Heidelberg, Germany. His current research interests include network based mobility protocols and extensions, with active contributions to standards (IETF and 3GPP), European projects, and internal business unit projects. Prototype implementation of the designed protocols is also one of his main activities.

Frank Zdarsky (fzdarsky@redhat.com) is a principal software engineer at Red Hat, responsible for the NFV/SDN technology and standards strategy. He has been active in ETSI NFV and OPNFV from their inception, and received the ETSI NFV Excellence Award for his contributions to the field. Prior to Red Hat, he was the head of NEC's European mobile network research, working on radio access and backhaul networks, and mobile core to service delivery platforms. He holds a Dr.-Ing. degree in computer science and a joint master's equivalent (Dipl. Wirtsch.-Ing.) in business administration and electrical engineering. He has more than 30 publications and an h-index of 13.

Marco Liebsch (marco.liebsch@neclab.eu) is currently working as a senior researcher at NEC Laboratories Europe in the area of mobility management, mobile content distribution, mobile cloud networking, and software defined networking. He has worked in different EU research projects and is contributing to standards in the IETF and 3GPP. For his thesis on paging and power saving in IP-based mobile communication networks, he received a Ph.D. from the University of Karlsruhe, Germany, in 2007. He has a long record of IETF contributions as well as RFC, journal, and conference publications.

Tarik Taleb [S'04, M'05, SM'10] (tarik.taleb@aalto.fi) received the B.E. degree (with distinction) in information engineering and the M.Sc. and Ph.D. degrees in information science from Tohoku University, Sendai, Japan, in 2001, 2003, and 2005, respectively. Prof. Taleb is a professor at the school of electrical engineering, Aalto University, Finland. He was a senior researcher and 3GPP standardization expert with NEC Europe Ltd. He was then leading the NEC Europe Labs Team, working on research and development projects on carrier cloud platforms. Prior to his work at NEC, he worked as an assistant professor at the Graduate School of Information Sciences, Tohoku University. His current research interests include architectural enhancements to mobile core networks, mobile cloud networking, mobile multimedia streaming, and social media networking. He has also been directly engaged in the development and standardization of the evolved packet system as a member of 3GPP's System Architecture Working Group. He is an IEEE Communications Society (ComSoc) Distinguished Lecturer. He is a board member of the IEEE ComSoc Standardization Program Development Board. He is serving as the Chair of the Wireless Communications Technical Committee, the largest in IEEE ComSoC. He founded and has been the General Chair of the IEEE Workshop on Telecommunications Standards: From Research to Standards, which is a successful event that received the "Best Workshop Award" from IEEE ComSoC. He is/was on the editorial board of *IEEE Transactions on Wireless Communications*, *IEEE Wireless Communications Magazine*, *IEEE Transactions on Vehicular Technology*, *IEEE Communications Surveys and Tutorials*, and a number of Wiley journals. He has received many awards, including the IEEE ComSoc Asia Pacific Best Young Researcher award in June 2009. Some of his research work has also received Best Paper Awards at prestigious conferences.