Laghrissi, Abdelquoddouss; Taleb, Tarik

A Survey on the Placement of Virtual Resources and Virtual Network Functions

Published in:
IEEE Communications Surveys and Tutorials

DOI:
10.1109/COMST.2018.2884835

Published: 01/01/2019

Document Version
Peer reviewed version

Please cite the original version:
Laghrissi, A., & Taleb, T. (2019). A Survey on the Placement of Virtual Resources and Virtual Network Functions. IEEE Communications Surveys and Tutorials, 21(2), 1409 - 1434. [8556457].
https://doi.org/10.1109/COMST.2018.2884835

A Survey on the Placement of Virtual Resources and Virtual Network Functions

Abdelquoddouss Laghrissi and Tarik Taleb

***Abstract*—Cloud computing and network slicing are essential concepts of forthcoming 5G mobile systems. Network slices are essentially chunks of virtual computing and connectivity resources, configured and provisioned for particular services according to their characteristics and requirements. The success of cloud computing and network slicing hinges on the efficient allocation of virtual resources (e.g. VCPU, VMDISK) and the optimal placement of Virtualized Network Functions (VNFs) composing the network slices. In this context, this paper elaborates issues that may disrupt the placement of VNFs and VMs. The paper classifies the existing solutions for VM Placement (VMP) based on their nature, whether the placement is dynamic or static, their objectives, and their metrics. The paper then proposes a classification of VNF Placement (VNFP) approaches, first, regarding the general placement and management issues of VNFs, and second, based on the target VNF type.**

***Index Terms*—NFV, Cloud, Network Slice, 5G, Mobile, and VNF Placement.**

I. INTRODUCTION

Slicing is the general term used when discussing virtualization techniques utilized to architect, partition and organize the computing and communication resources of a physical infrastructure to enable flexible support for diverse use cases. This partitioning of resources is meant to be optimized for a specific requirement and/or a specific service in a cost-efficient manner, and to answer to the diverse requirements of emerging 5G verticals/applications. Network slicing, believed to be the key ingredient of 5G and beyond networks, consists of allowing a multitude of logical networks to be created on top of a common physical infrastructure, and to share its resources, by turning traditional structures into customizable elements that can run on the architecture of choice [174]. Effectively, in 5G, it is anticipated to have a slice dedicated to streaming services and another dedicated for social media services, jointly running on top of a shared physical infrastructure. A logical network slice, in our case, is considered mainly as a logical combination of network functions and virtual resources, regardless of the resource isolation among the tenants.

To enable such logical network slices and to accommodate several 5G use cases (e.g., mission-critical applications, media personalization, and mobile broadband), the virtualization capabilities offered by NFV will benefit many industries. The fast deployment and simple management feature, dynamicity, and high availability of VNFs [159] effectively enable the provisioning of a smart segmentation, customization, and programmability of the network to meet the needs of each service. These VNF properties depend on several factors, such as whether they are running over Virtual Machines (VMs) or containers (i.e. running with DevOps-style for additional flexibility and efficiency of service applications [160]).

Indeed, the evolution towards 5G consists of managing highly dynamic network slices consisting of several virtual nodes. They can be created or destroyed depending on service requests, or any objectives defined by mobile operators, such as cost reduction (e.g., capital and operational expenses – CAPEX/OPEX) and energy consumption. The need for network slices, which will enable operators to provide networks "in an As A Service (AAS)" fashion, proves itself to be the key concept for future use cases, such as putting both bandwidth and latency demands on the network, defining optimal personalized verticals to answer, in a dynamic and flexible manner, to the requirements of users, specific applications, and services [121], [162]–[164], [167], [168].

The pressing need to customize specific applications and services, according to the preferences and behaviors of end-users in consuming the services, has motivated a large library of research work. Several architectures, combining cloud computing and mobile networks, have been proposed in the recent literature [25], [77], [78]. A highly dynamic network management architecture is introduced in [69], whereby both nodes and links are virtual. This architecture is based on an orchestrator which carries out automatically the placement of nodes depending on a system that collects information about the resource consumption. The real challenge is to produce efficient and scalable software for managing and orchestrating virtual networks of the future. These virtual networks need to be configured and have their life cycles managed. Besides, the elements of virtual networks need to be allocated in a dynamic manner on physical machines by using efficient resource allocation and VNF placement algorithms. In the same fashion, a novel concept dubbed "Follow Me Cloud" (FMC) is proposed in [45]. It allows services to migrate and seamlessly follow the mobility of users by selecting Data Centers (DC) based on the delivery rates in the network and the locations of users. The main idea of FMC is that services follow users throughout their movement. Two of the key technologies to realize such a concept are Software Defined Networking (SDN) and virtualization. The virtualization offers

A. Laghrissi and T. Taleb are with the Department of Communications and Networking, School of Electrical Engineering, Aalto University, 02150 Espoo, Finland. T. Taleb is also with the Centre for Wireless Communications (CWC), University of Oulu, FI-90014 Oulu, Finland, and the Computer and Information Security Department, Sejong University, 143-747 (05006) Seoul, South Korea (emails: firstname.lastname@aalto.fi).

TABLE I: Acronyms used in this paper.

Acronym	Meaning
AAS	As A Service
ACO	Ant Colony Optimization
ADC	Application Delivery Controller
CAPEX	CAPital EXpenditure
CDN	Content Delivery Network
CDNaaS	Content Delivery Network as a Service
Cloud SP	Cloud Service Provider
CSP	Constraint Satisfaction Problem
DES	Double Exponential Smoothing
DC	Data Center
DInf-UFPR	Department of Informatics of Federal University of Parana
DIP	Direct Integer Programming
DN	Data Node
DNA	Digital Network Architecture
DPI	Deep Packet Inspection
EPC	Evolved Packet Core
ETSI	European Telecommunications Standards Institute
FFA	First Fit Algorithm
FMC	Follow Me Cloud
GA	Genetic Algorithm
HGA	Hybrid Genetic Algorithm
HSS	Home Subscriber System
HwPFA	Hardware Predicted Failure Analysis alerts
IaaS	Infrastructure as a Service
ILP	Integer Linear Programming
IoT	Internet of Things
ISP	Internet Service Provider
IT	Information Technology
LOPI	Local Optimal Pairwise Interchange
MC	Markov Chain
MCC	Minimum Correlation Coefficient
MGAP	Multi-level Generalized Assignment Problem
MGGA	Multi-objective Grouping Genetic Algorithm
MIP	Mixed Integer Programming
MME	Mobility Management Entity
MODM	Multiple Objective Decision Making
NSGA	Non-dominated Sorting Genetic Algorithm
NFV	Network Function Virtualization
NFVI	Network Function Virtualization Infrastructure
NFV MANO	Network Function Virtualization Management and Orchestration
OPEX	OPerational EXpenditure
OS	Operating System
OVMP	Optimal Virtual Machine Placement
PABFD	Power Aware Best Fit Decreasing
PBO	Pseudo-Boolean Optimization
PBFVMC	Pseudo-Boolean for Virtual Machine Consolidation
PGW	Packet data network GateWay
PM	Physical Machine
PoD	Point of Delivery
PSO	Particle Swarm Optimization
QAP	Quadratic Assignment Problem
QoS	Quality of Service
RAN	Radio Access Network
RLWR	Robust Local Weight Regression
ROI	Return On Investment
SA	Simulated Annealing
SDN	Software Defined Networking
SFC	Service Function Chaining
SGW	Serving GateWay
SKF	Simple Kalman Filter
SIP	Stochastic Integer Programming
SLA	Service Level Agreement
TVMP	Traffic-aware Virtual Machine Placement
TVPR	Time-aware Virtual Machine Placement and Routing
UE	User Equipment
VBP	Vector Bin Packing
VDI	Virtual Desktop Infrastructure
VIM	Virtual Infrastructure Manager
VM	Virtual Machine
VMP	Virtual Machine Placement
VMPACS	Virtual Machine Placement Ant Colony System
VMPDN	Virtual Machine Placement for Data Nodes
VMcP	Virtual Machine consolidated Placement
VMiP	Virtual Machine incremental Placement
VNF	Virtualized Network Function
VNFP	Virtualized Network Function Placement
VNFaaS	Virtual Network Function as a Service
VNFC	Virtualized Network Function Component
VNF-FG	VNF Forwarding Graph

the ability to change the location of a VM or a container from a given host to another without interruption, leveraging SDN, and with a small impact on the network performance.

Appliances based on dedicated hardware are limited in terms of scalability and do not easily support the prompt launch of new services. VNFs have helped in the acceleration of service provisioning and innovation. VNFs were first defined in [1], as software implementations of network functions that can be deployed on a Network Function Virtualization Infrastructure (NFVI), enabling the agility of networks to automatically respond to the needs of the traffic and services running over it. NFV is a new model (Fig. 1) that stands for running VNFs (i.e., software components of network functions) on standard VMs. With NFV, network functions become software-based, multiple and diverse roles can take place over the same hardware, networks become remotely and dynamically configurable particularly with the help of SDN, and the overall network architecture/service delivery platform becomes easily scalable. Besides NFV, SDN enables inter-working of these network functions, whether they are launched on different VMs in the same DC or across multiple DCs, to obtain a mobile, flexible, and dynamic network that is rapidly deployable in the cloud [53]–[55], [70], [84], [101]. The dynamic nature of VNF Placement, to form such network slices, despite its numerous benefits, may result in sub-optimal or unstable configurations of virtual networks if not chosen wisely. Furthermore, it is critical to express the mobile services requirements and the state of the network infrastructure to define the different placement constraints and to obtain a viable configuration. So far, mobile networks, mainly serving cell phones, have been optimized for phones only. However, in the 5G era, they have to serve a variety of devices, associated with diverse verticals, with different characteristics and needs. Some of the typical use cases of 5G are mobile broadband, Internet of Things, and Autonomous Driving. They all exhibit different features and have different requirements regarding mobility, latency, reliability, etc. [120]. Creating optimal network slices for each 5G service/vertical largely hinges on efficient algorithms for the placement of relevant VNFs along with mechanisms for the allotment of corresponding virtual resources.

To the best knowledge of the authors, there is no extensive survey, in the literature, on the problem of VNF placement in cloud environments, apart from the work presented by Li et al. in [119]. In this work, the authors raised some questions related to the interoperability of network functions, the origins

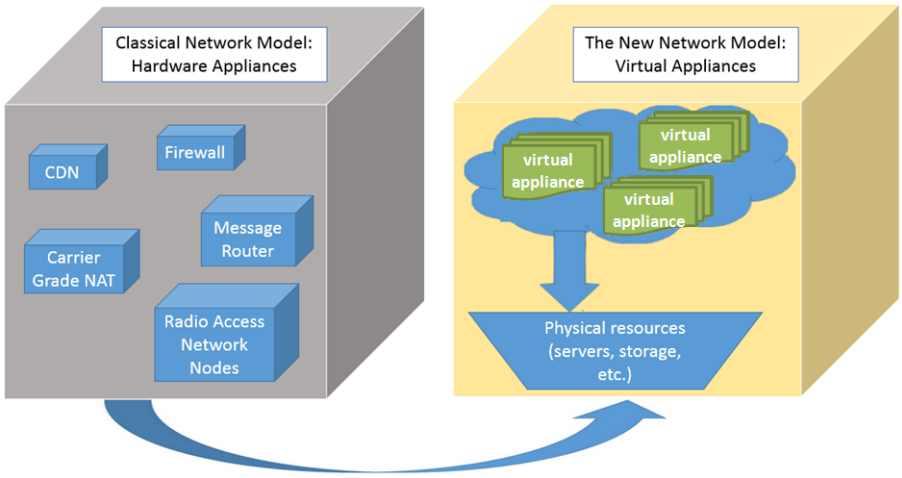


Fig. 1: Trendy shift towards the virtualization of networks and their functions.

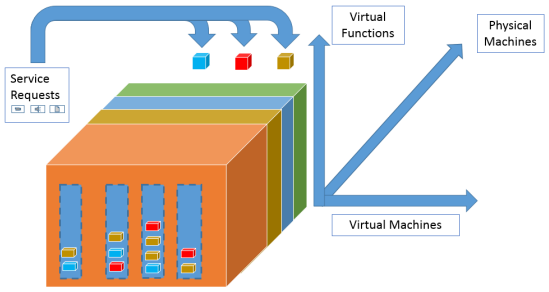


Fig. 2: The problem of network function mapping and placement.

of service chains and the On-path placement of network functions. Through these questions, their survey addresses the placement issues of both conventional dedicated hardware-based network functions and VNFs that are currently popular. In comparison to the survey of Li et al., this survey is more extensive and detailed as it discusses in depth the existing VNF placement strategies and algorithms. It also classifies the different solutions into different categories, distinguishing between generic VNF placement and specific VNF placement. These categories are:

- Network function chain placement which dynamically steers the traffic through an ordered list of Service Functions (SFs), mainly located in a middle-box (e.g., Firewall and Deep Packet Inspection - DPI), and facilitates the dynamic enforcement of service-inferred traffic forwarding policies [169].
- VNF forwarding graph which defines a graph of interconnected VNFs which are linked in order to instantiate a Network Service.
- VNF replications which create replicas of a VNF or a set of VNFs (i.e., Service Function Chaining - SFC - with replications) in order to provide load balancing and recovery capabilities for the network.

As shown in Fig. 2, studying the VNF placement problem can be done through the study of the orchestration, management, and configuration of specific VNFs or through considering the placement of VMs [79], their management [138] and linkage to User Equipment (UE) mobility and service usage [139]. In this vein, this paper thoroughly explores both approaches. The paper will present some interesting VNF and VM use cases and the issues that are relevant to their placement. To explore the concepts related to NFV and its architecture, the interested reader may refer to [131].

To help the reader grasp the relationship between VMs and VNFs, and consequently the relationship between VMP and VNFP, we depict in Fig. 3 an example of the mapping between VNFs and VMs in case of an OpenStack Infrastructure as a Service (IaaS), and that is through the VNF Manager (VNFM) which instantiates, scales up/down, updates, and terminates VNFs; the Virtual Infrastructure Manager (VIM), which is responsible for controlling and managing the NFVI compute, storage, and network resources; also a VNF includes Virtual Deployment Unit(s) (VDUs) which is the VM hosting the network function, the connection point(s) connecting the internal virtual links or outside virtual links, and the virtual link(s) which provide connectivity between VDUs.

The remainder of this paper is organized as follows. Section II and Section III include the definitions and use cases of VMs and VNFs, respectively. Section IV discusses the VMP problem and classifies the related work. Each class of VMP solutions is introduced in a separate section, namely Section V for energy consumption minimization, Section VI for cost optimization, Section VII for Quality of Service (QoS), Section VIII for resource usage, Section IX for reliability, and Section X for load balancing. The paper introduces the general VNF placement approach and its related issues in Section XI. In Section XII, we discuss the work dedicated to specific VNF types. Finally, Section XIII presents the key challenges and lessons learned, and Section XIV concludes the paper, offering a recap on important areas and highlighting open research areas.

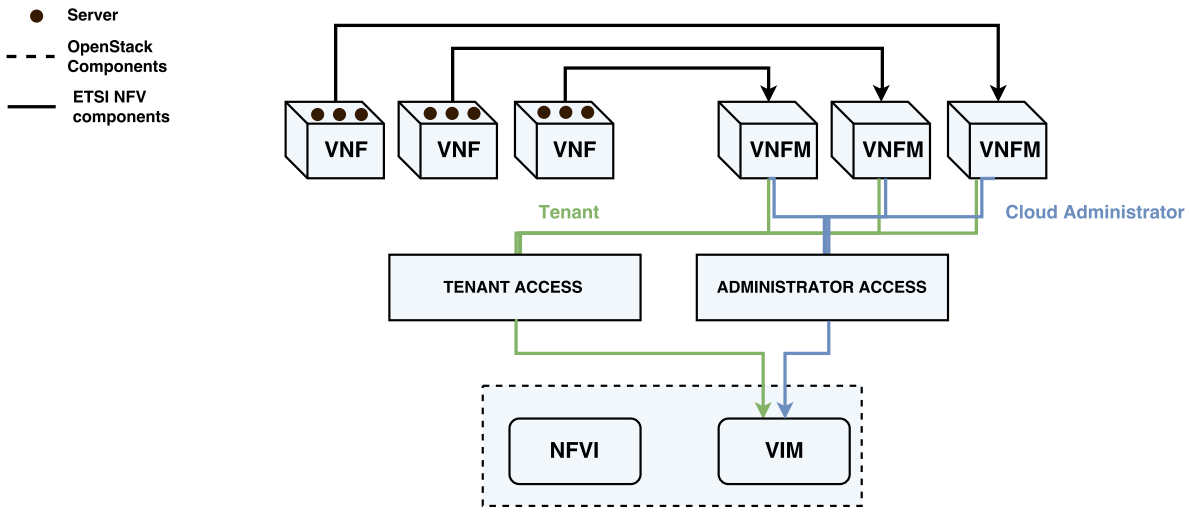


Fig. 3: Example of mapping of VNFs and VMs on an OpenStack IaaS.

II. Virtual Machines

With the evolution of virtualization technologies, the use of VMs has become more common to perform some tasks in a way different than if it was implemented in a physical machine (PM). VMs, being basically a software implementation emulating the behavior of a computing environment wherein programs/operating systems (OSs) can be installed and run, are implemented by means of hardware virtualization techniques. Indeed, the virtualization stands for creating virtual resources on-demand with the main objective of managing different workloads and making traditional computing more scalable. VMs can be seen as sets of data files which can be moved/copied from one PM to another. Unlike the hardware-level virtualization provided by VMs, containers share the kernel of a given host's system with other containers, which constitutes an OS-level virtualization. So, while VMs run applications on their respective guest OS, on top of a single hypervisor (which runs on top of the host OS), containers run on top of, for instance, one common Docker engine, running on top of the host OS. The open-source Docker uses the kernel features of Linux relative to control groups, namespaces and the creation of containers on top of an OS. In this section, we will introduce some of the most relevant VM use cases: VM lifecycle management, VM migration, and containers.

A. VM use cases

Many use cases have been defined for VMs to make the best use of virtualization systems, namely to enhance workload handling measures, backup, and migration. It is therefore important to define the steps to follow in order to make the best use of a VM so that it efficiently accomplishes a given task. In this section, we define several VM use cases.

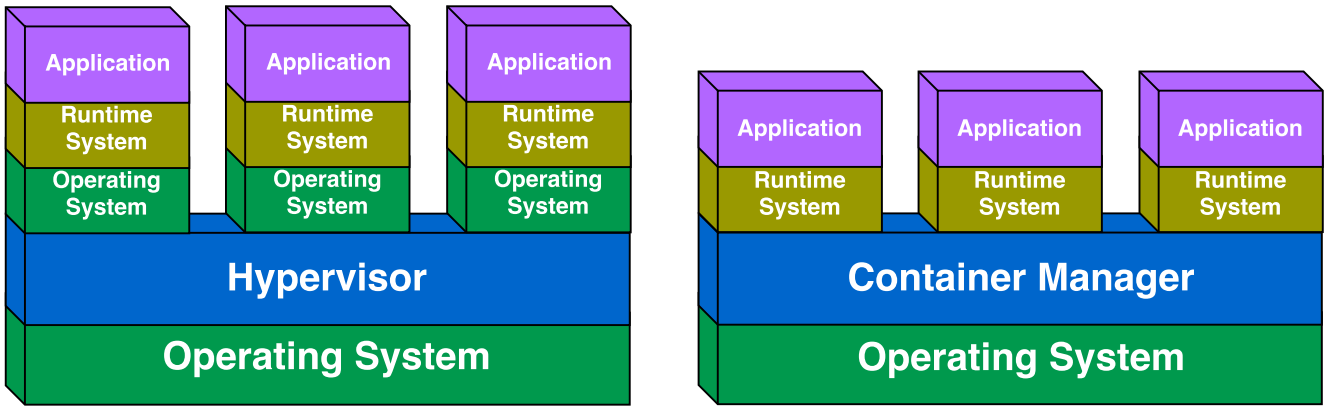
1) VM Lifecycle Management: Several virtualization products describe the use cases related to VM lifecycle management, defining a set of operations to help administrators to supervise the implementation, operation, and maintenance of VMs. The objectives of such use cases are to support the full VM state management, to define a unified approach for the management of virtual and physical servers, to monitor VM health and assets, and to enable automatic policy association. A number of tools implementing such objectives are offered by several vendors such as VMware Inc, VDIworks, and Virtual Computer Inc.

2) Virtual Machine Migration: The inability to migrate physical servers and the implications incurred (i.e., on availability and failure recovery) has motivated the migration capabilities of VMs within and across servers/data centers. The main use cases for moving VMs are:

- Achieving better performance by moving VMs from one location to another, for example by avoiding busy servers and ensuring load balancing.
- Moving VMs from servers which need upgrades, maintenance, or any other operation that could take place in normal hours rather than overnight or during weekends.
- Achieving high availability by instantiating VMs on alternative servers when their current physical servers are failing or get inadvertently down.
- Replacing physical servers with no downtime by migrating VMs to other servers. For example, Vsphere offers various migration mechanisms which support such use cases.

3) Containers in Virtual Machines: First, it is important to clear up the ambiguity between VMs and containers, as they may seem similar at different levels of granularity. Both are meant to ensure application isolation (including the isolation of applications' dependencies) into an independent and self-contained unit capable of running anywhere. However, they are different mainly in their architectural approaches as depicted in Fig. 4.[1] Also, VMs emulate "real" PMs while running on top of host machines using the hypervisor. The hypervisor has the main task of provisioning VMs with i) a platform to execute

[1]An "application", running on top of VMs or containers, can be deemed as a VNF.



(a) VM's architecture

(b) Container's architecture

Fig. 4: VM's architecture versus container's architecture.

and manage the OS of the guest VMs, and ii) the resources shared by the host machines amongst the VMs running as guests on top of them. Despite this, many organizations embrace the combination of containers and VMs for applications with the workload that is suitable for containers' platforms. In such environments, several application-level use cases concern containers running inside VMs:

- Stateful application migration: With the advantage of low resources consumed by containers, stateful applications are known to rely on redundant infrastructures that can be migrated, e.g., in case an error occurs in the hardware hosting the VM on which the given container is running.
- Cloud portability: Containers allow the deployment of applications on several distributions, regardless of the installed packages and the container type. This heterogeneous distributions and platform choices allow a public cloud portability for the containerized applications. Despite the many advantages this could bring, several dependencies should be taken into consideration, such as which applications are more suitable to run in a container versus a VM, computation resources maximization requirements, security, maintenance, and sprawl avoidance. In this paper, the focus is on VM placement as there are not many solutions for container placement in the literature. However, many VM placement approaches can be adapted to also optimally place containers.

III. Virtualized Network Function as a main component of Network Function Virtualization

The European Telecommunications Standards Institute (ETSI) established the concept of Network Function Virtualization (NFV) and defined the basic architecture and requirements of VNFs [4]. The NFV framework consists of three main components: VNFs, NFVI, and the NFV Management and Orchestration (NFV MANO). Along with a plethora of devices, ranging from smart phones (and soon intelligent phones) to IoT sensors – consuming variant applications, and using different high-speed transmission technologies – the digital tsunami cannot be handled by traditional methods and existing solutions [172], [175]. There is therefore a need to study additional new use cases, other than those related to VMs.

In this vein, several solutions have been proposed based on NFV, such as "Enterprise Network Function Virtualization (E-NFV)" which is the key component of the "Cisco Digital Network Architecture (DNA)". This solution helps Information Technology (IT) teams working in networking companies to handle the security and complexity management issues they may face. Indeed, the Enterprise NFV Design is one of the most appealing use cases of NFV. Such a use case is just one of many, as NFV has opened up new ways of making progress towards simple, agile and programmable networks which can handle the tendencies of new technology [2].

Indeed, NFV enables the elastic scaling and rapid deployment of network functions, replacing the need to set up and maintain the correspondent hardware such as firewalls, gateways, and transcoders. The provision of such VNFs over virtualized infrastructures defined several use cases and system requirements. ETSI has selected a set of relevant ones, such as the Virtualized Network Functions as a service (VNFaaS), the Virtualization of mobile base stations, and the virtualization of Content Delivery Networks (CDNs). In this section, we will introduce the main NFV use cases [3], [5], [6].

1) Virtual Network Function as a Service: The outsourcing, management and deployment of virtual network layers for service providers will profit IT companies. The management of virtual networks, globally and in a distributed environment, requires that they scale up and down automatically, which many IT companies cannot afford. Despite this, just a few solutions for security and Application Delivery Controllers (ADC) are embedding VNFaaS in their deployments of NFV [154]–[156].

2) Network Function Virtualization as a Service: NFV came with the promise to reduce costs (i.e., capital expenditures (CAPEX) and operational expenditures (OPEX) which define the cost or charge for operating a system) and increase profits. To attain these objectives, communication service providers and Cloud Service Providers (Cloud SPs) are working on improving their IT infrastructures. They are expected to go beyond VNFaaS to offer a whole NFV infrastructure as a service which will result in the expansion of network service

classes and types.

3) Service Function Chaining: In hybrid environments, where VNFs and hardware appliances provide services jointly, SFC is an emerging architecture which permits the establishment of simple configurations that make it easier for a Network Service Provider to manage and enforce several policies related to access control, security, QoS, etc. SFC is very important for the granular management of virtual networks and will require the usage of VNF forwarding graphs (VNF-FGs). This will be eminently required due to the increasing number of deployed VNFs and QoS-sensitive services as well as the needed maintenance of point-to-point inter-VNF connections.

4) Virtualization of Mobile Core services: To accelerate 5G and to support flexible, rapid and reliable deployment of more mobile network services, the underlying infrastructure will be improved by virtualizing the mobile core services. We are already witnessing the use of the virtual IP Multimedia System (vIMS) and virtual Evolved Packet Core (vEPC) within NFV frameworks. Also, it resulted in enhancing costs and speeding the time of service to market. The virtualization of mobile core functions will also provide the ability to deploy cost-effective network services even when reaching rural areas.

5) Virtualization of Content Delivery Networks: VNFs will allow service providers to provision the amount of dedicated networks for optimal multimedia traffic delivery, all in the same network wherein they deliver every other service traffic. Thus, there will be no need to subcontract with multimedia service providers. Ultimately, NFV can be the key to solve many issues that could disrupt the functioning of CDNs [140]–[142], [176].

6) Home and Business Gateways virtualization: Internet Service Providers (ISPs) count on embedded processors-based set-top boxes and residential gateways. VNFs running on processors will replace the physical infrastructure consisting of processors and an Application-Specific Integrated Circuit (ASIC). This is much more cost-effective and does not need high bandwidth to deploy. These virtual Customer Premises Equipment (vCPE) implementation will shed light on white boxes which are more agile and with a lower-cost. It will also present a universal platform whereby VNF services (e.g., optimization or security services) can be deployed on-demand.

IV. Virtual Machine Placement

As stated earlier, since VNFs run on top of VMs, it is important to provide solutions that effectively plan the provisioning of VMs with respect to the needed SLA requirements. With regard to this matter, this section introduces and classifies the different research work dedicated to the placement of VMs.

VMP is the selection process of the appropriate physical hosts in cloud DCs to instantiate new VMs. This selection process can be carried out either offline (static) or online (dynamic) [135]–[137]. In the case of the offline VMP approach, DC operators gather inputs and make placement decisions to satisfy requests from multiple end-users taking into account different constraints. In the online VMP approach, in addition to the placement decisions, DC operators periodically collect data and decide, e.g., when the load of the system increases, whether a VM placement shuffle is needed.

Ongoing advances in virtualization technologies do not only allow sharing resources among several VMs, but also migrating VMs from one physical host to another, most importantly, without service interruption [10]. This migration is present in many technologies such as VMware ESX [11] and Xen [8]. It considers different constraints relevant to compatibility at the virtualization level (i.e., virtualization software) as well as at the infrastructure level (i.e., CPU, RAM, etc.). This has motivated much research work dealing with the mapping and placement of VMs.

TABLE II: Global classification of VMP solutions.

Type of placement	Mono-objective	Multi-objective
Online VMP	[12] [16] [22] [26] [29] [30] [33] [35] [38] [40] [44] [52] [63] [65] [66] [83]	[20] [21] [23] [27] [36] [46] [49] [50] [51] [64] [94][95] [100]
Offline VMP	[17] [24] [32] [34] [37] [39] [47] [48] [62] [68] [80] [81]	[13] [23] [27] [31] [51]

TABLE III: Objective-based classification of VMP approaches.

	Objective	Type of placement	Reference
Energy-aware VMP	Power consumption	Online	[38][51][63] [65] [95]
		Offline	[13] [37] [47] [48] [52] [62] [80]
	Number of active nodes	Online	[20] [30] [46] [64][66] [83] [84] [94]
		Offline	[66] [68][17] [34] [39]
Cost-aware VMP	Operating cost	Online	[16][23][27]
		Offline	[23][27]
	User's budget	Online	N/A
		Offline	[32]
	ROI	Online	[50]
		Offline	[31]
	Power budget	Online	N/A
		Offline	[31]
QoS-aware VMP	Overhead	Online	[29]
		Offline	N/A
	Congestion	Online	[20] [36] [64]
		Offline	[81]
	Aggregate Traffic	Online	[23][27]
		Offline	[23][27]
	Data transfer time	Online	[22]
		Offline	N/A
	Delay	Online	[64] [94]
		Offline	N/A
	Latency	Online	[35] [40] [52] [67]
		Offline	N/A
Resource usage		Online	[21] [26] [36] [44] [46] [51] [49] [95] [100]
		Offline	[13] [51]
Reliability		Online	[21] [33] [100] [50] [94]
		Offline	[24]
Load balancing		Online	[[12] [49] [50] [66]
		Offline	[51]

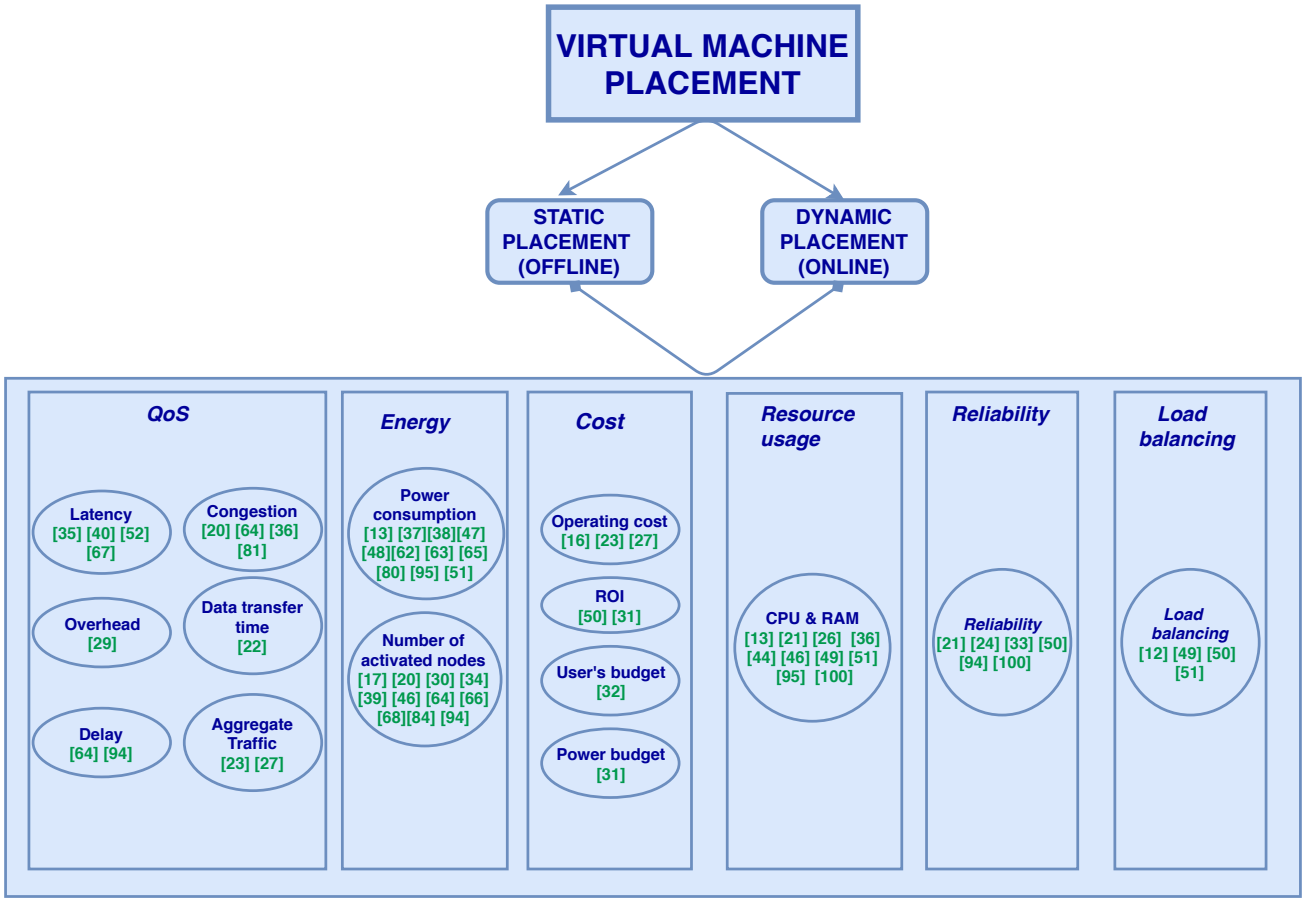


Fig. 5: VMP classification.

Table II shows a global classification of VMP solutions into online and offline approaches, while Table III categorizes the different VMP solutions as per their target objective. Some of the solutions are dedicated to single objectives (mono-objective), and some have several objectives (multi-objectives) (see Fig. 5). Among the objectives are the following:

- Energy consumption minimization: translated by the minimization of power consumption and the number of active nodes.
- Cost optimization: can be expressed in terms of the Return On Investment (ROI), resource exploitation cost or the VM allocation cost.
- QoS optimization: can be expressed in terms of response time, overhead time, etc.
- Resource usage: RAM, CPU, storage, etc.
- Load balancing: the avoidance of congestion, data overload, etc.

Each of these objectives will be discussed in the following sections (Sections V to X). Also, for each section, we summarize the challenges and suggestions, concerning the most relevant/recent research works, in a dedicated table (Tables IV to IX). Each table contains the advantages and disadvantages that we have assessed for the adopted solutions and frameworks, as well as enhancement propositions that could guide the reader to spot possible research directions.

V. Energy-aware VMP

For an energy-efficient VMP, the general approach considers reducing the number of powered ON PMs or minimizing the power consumption, and this is carried out through policies that are compliant with the Service Level Agreement (SLA).

A. Power consumption

1) Online: As an improvement to a previous Genetic Algorithm (GA) solution in [38], a Hybrid GA (HGA) is introduced by Tang et. al. in [63]. It incorporates a procedure for local optimization and one for repairing infeasible solutions. These enhancements exploit the capacity and convergence of the previous solution. HGA takes into account the constraints of required main memory and total CPU. The procedures re-allocate VMs, which violate those constraints, to other PMs until no violation remains. The HGA exhibits better performance and efficiency than the original GA. A self-adaptive placement strategy, based on Robust Local Weight Regression, is proposed by Zhang et. al. in [65]. With the goal to retrieve a compromise between energy consumption and SLA, this approach aims at dynamical changes of workload requirements, deciding the overload time of hosts dynamically. Considered as the most energy-consuming component, the model focuses on the power consumption of the CPU. Its operations consist of selecting VMs, detecting overloaded

hosts and migrating the necessary VMs to the underloaded hosts. The cloud experimentation environment includes a DC that contains a given number of heterogeneous nodes, and the results show that the solution algorithm can complete dynamic VMs consolidation and significantly reduce energy consumption while adhering to the SLA requirements. Consolidation refers to when data storage (i.e. Storage consolidation) or server resources (i.e. Server consolidation) are shared among multiple users and accessed by multiple applications in order to avoid the underutilization of resources.

2) Offline: In cloud computing, VM consolidation proved itself as an efficient way to save energy. Nevertheless, the need to provide good service quality makes it necessary to find a fair tradeoff between energy saving and performance. To solve this issue, Ribas et al. in [37] introduced an artificial intelligence approach based on a Pseudo-Boolean (PB) formulation. Although the proposed solution shows better consolidation results compared to the First Fit Algorithm (FFA), the experiments using the data center of the "Informatic Department of Federal University of Parana (DInf-UFPR)" and Google Cluster show their limitations when applied to large realistic data. To cope with these limitations, an improved Pseudo-Boolean Optimization (PBO) of the VM consolidation problem, called "PBFVMC" is proposed in [47]. With the same objectives, many variables are taken into account, such as the amount of RAM, processing power, and running hardware type. The considered constraints (e.g., the necessary amount of ON resources to power all VMs, the hardware on which a VM is running and must be ON, etc.) led to a significantly high number of variables (i.e., $(2 \times N + 2 \times N \times K)$ variables for $(2 + 2 \times N + K)$ constraints, whereby K and N represent the number of available VMs and hosts, respectively). The conducted experiments used a data trace from the Google Cluster project. They show that in spite of the large number of variables and constraints, the new approach can decrease the number of variables by 50% and execute huge sets of VM instances leading to a shorter execution time and better consolidation results.

A "redesigned energy-aware heuristic framework for VM consolidation to achieve a better energy-performance tradeoff" is proposed by Cao et al. in [62]. The framework, as a redesign of CloudSim, classifies the overload in the host status into two types: either with or without SLA violation. Then, a minimum power and maximum utilization heuristic makes the energy-aware VMP decisions. The conducted experiments and the performance evaluation show that the proposed solution outperforms the original framework, significantly decreasing the consumed energy, execution time, and SLA violations. With the same objectives of decreasing energy consumption and SLA violation, Fu et al. introduced a new model for energy consumption and cost of VM migration, as the basis of an improved VM selection policy [80]. This policy, inspired by the "Power Aware Best Fit Decreasing (PABFD)" and called the "Minimum Correlation Coefficient (MCC)", is used to assess the level of association between a given VM and PM to avoid the performance degradation on other VMs. It reduces the SLA violation rate and consequently the power consumption. Using cloudSim-3.0, the policy is demonstrated to achieve better performance compared to the previous work, but it is not applied in a real environment.

In [48], a "multi-component utilization-based power model", proposed by Dalvandi et al., addresses the limitations of "time-aware VMP and Routing (TVPR)", whereby each user requests a number of VMs and a specific bandwidth amount for a given duration. The proposed model defines the energy usage of a cloud DC depending on the utilization of all its components. Based on this power model, a mixed integer linear optimization problem is formulated. It is solved using a "least-active-most-utilized policy" solution. As objectives, the solution aims at minimizing the total power consumption and maximizing the number of accepted demands, while taking into account the constraints of capacity, flow conservation, and demand satisfaction. The obtained acceptance ratio and power consumption for both small and large DCs prove the effectiveness of the solution.

B. Number of activated nodes

1) Online VMP to reduce the Number of activated nodes: The effective usage of electricity and hardware resources in the cloud, along with jointly satisfying users with a good quality of service, is a challenge that cloud providers are facing. The optimization of the number of active PMs can help cut down considerably the power consumption. In [20], Bellur et al. present two approaches based on linear programming and quadratic programming to derive near-optimal solutions for the problem. The problem is seen as a Vector Bin Packing problem (VBP), the objective of which is to minimize the number of PMs. Compared to the existing theoretical worst-case bound for the VBP problem, the solution named "Packing Vectors" gives near-optimal solutions, although the dynamic placements are not handled to meet efficiently the typical workload of modern applications [98].

In [83], Moorthy proposes a VMP scheme based on two components, namely a VM monitor and a resource provisioner with the objective of minimizing the number of PMs used. Those two components are based on a Constraint Satisfaction Problem (CSP), which considers the completion time. The objective of the resource provisioner is to choose an optimal PM for a VM to satisfy the given demand constraints, and to choose a resource with the minimal completion time. Once the VM is placed, the VM Monitor keeps on monitoring the CPU usage of the PM where the VM is hosted and migrates the VM if the PM is found to be overloaded (i.e., if the CPU usage of the PM exceeds a given threshold). The performance of the scheme is better compared to the first fit algorithm, regarding completion time, user satisfaction and the number of created machines.

The load placement policies can play a major role in reducing the energy consumption for DCs. It has been also successfully demonstrated that they have an impact on cooling down the maximum DC temperatures, mainly, for service providers that manage multiple geographically distributed DCs. In [30], Le et al. propose dynamic load distribution policies with migration that provide predictions of future

TABLE IV: Analysis summary of the relevant/recent work on energy-aware VMP solutions.

Ref	Objectives	Constraints	Algorithm(s)/ Approach(es)/ Policy(ies)	Advantages	Disadvantages	Suggestions to enhance the proposed solutions
[13]	Power consumption	CPU and RAM	- Bin-packing - Simulated annealing	In a centralized environment, the solution can be applied to improve the energy consumption of the ensemble layer.	- Added complexity in the system when using the proposed solutions. - The experimentation was conducted on a centralized environment. - Cannot know if it will be performing well in a federated environment. - The energy evaluation is assessed for the overall system.	- Could consider using Distributed Management Task Force standards. - Consider energy consumption on the granularity of each DC, including, disks, memory, etc.
[30]	Number of active nodes	SLA run-time, cost	Dynamic load distribution policies	Good results in the experimentation environment.	- Tested only on a small-scale environment. - Since the load distribution is carried out on a per-service request basis, its applicability for real cloud services/datasets with a large number of requests is questionable.	Study of the applicability of such an approach in datasets with a very large number of service requests.
[80]	Power consumption	CPU utilization	Power Aware Best Fit Decreasing (PABFD)	Outperforms the original framework within CloudSim and its enhanced version proposed in [63].	Even if the policy is demonstrated to achieve better performance compared to the previous work, it is not however tested in a real cloud infrastructure.	Extend it to be tested on a real cloud environment such as OpenStack.
[83]	Number of activated nodes	Service demand and minimal completion time	Constraint Satisfaction Problem	The way the conflicting objectives are handled in the proposed CSP solution makes it easier to achieve optimal solutions.	Since CSP explores all possible solutions for a set of input data, it cannot be applied to very large datasets.	Application of constraint propagation to reduce the number of possible values of each decision variable.
[94]	Number of active nodes, and reliability	The cloud resources consumption of an end-user, the response time, maximum tolerable failure rate	Constraint Satisfaction Problem with choco solver	The way the conflicting objectives are handled in the proposed CSP solution makes it easier to achieve optimal solutions.	Since CSP explores all possible solutions for a set of input data, it cannot be applied to very large datasets.	Application of constraint propagation to reduce the number of possible values of each decision variable.

migrations in a cost-aware fashion to pre-cool the DCs. The approach saves cost, respects the SLA run-time constraint, and makes placement decisions for each arriving service request. The placement decisions rely on DCs with the minimum necessary number of active servers and least cost.

2) Offline VMP to reduce the Number of activated nodes: Hieu et al. address in [66] the resource utilization among multiple resource dimensions, as a multi-dimensional VMP that considers multiple types of resources, namely memory, CPU, storage, and bandwidth. A VMP algorithm, named "Max-BRU", is proposed to balance the load across the defined resources by maximizing at the same time the resource utilization. Max-BRU determines the most appropriate physical server for deploying the VM requests based on metrics which relate to the least used host, in terms of the needed resources. Max-BRU makes an efficient use of these resources and reduces the number of required active physical servers in comparison with the greedy FFA proposed in [34], [39], the Load-aware policy used in [66], the VectorDot used in [14] and the market mechanism approach proposed in [17].

The fact that many VMP solutions focus on small-scale VMP schemes motivates the work carried out by Song et al. [68]. From the perspective of optimizing VM deployment, a new large-scale scheme based on the powerful convex optimization theory is proposed to reduce the number of PM deployments, decrease the communication cost between VMs and improve the energy-efficiency and scalability of DCs. In this scheme, an optimization-based algorithm considers the server-side constraints and application multi-tier inherent dependencies to make VMP decisions. Based on four network

experimentation topologies, namely Tree, VL2, Fat-Tree, and BCube, the proposed scheme saves considerably the traffic flow for the four topologies compared to the bin-packing algorithm.

VI. Cost related to the Cloud Service Providers' profit and virtual resources usage

Enabling the dynamic and automatic provisioning and placement of VMs is a challenging issue, mainly when considering application-level SLA requirements and resource exploitation [18]. In this section, we refer to the costs as the parameters that could impact the profit of Cloud SPs, namely the Return on Investment (ROI), the VM allocation cost, and the resource exploitation cost.

A. Online Cost-aware VMP

Within the decision layer, in [16], Van et al. rely on a two-level architecture separating the stage of VM provisioning from that of VMP. The first phase, referred to as VM provisioning, determines which VMs should be instantiated or destroyed. The second phase, referred to as VM packing, consists of placing the new VMs and deciding on any possible VM migration. The problem is formulated as CSP. The proposed CSP solver is based on Choco to implement the two phases as separate CSPs with the objectives of minimizing the number of active PMs and to maximize a given utility function. This function is calculated as a weighted sum of both the operating cost function and the utility functions of the application-provided resource-level. Regarding the nature of the CSP solver, known to be an exact method, the time allocated to find a solution is limited, leading to acceptable solutions.

To cope with the need of cloud providers to gain profit from SLA-compliant placement of VMs, an "SLA-aware placement of multi VM elastic services in compute clouds" is proposed by Breitgand et al. in [29]. The problem is presented as a multi-unit combinatorial auction and formulated as Direct Integer Programming (DIP). Compared to the column generation method, near optimal solutions are obtained by DIP combining reasonable time and good quality with the aim of maximizing the system availability and minimizing the network overhead due to VM migrations.

B. Offline Cost-aware VMP

Costs can be studied from the perspective of payment plans, along with dealing with the under-provisioning and over-provisioning problems of resource management in the cloud. To minimize the cost of hosting VMs in a multiple cloud provider environment under several demand and price uncertainties, Chaisiri et al. propose in [15] an algorithm named "Optimal Virtual Machine Placement (OVMP)". To get resources from cloud providers, OVMP is based on the solution provided by Stochastic Integer Programming (SIP). The experimentation results show that OVMP can minimize considerably the budgets of users when VMs are reserved. This plan is found to be cheaper than the on-demand plan but the necessary decisions to allocate the virtual resources with the exact amount needed by the users is difficult. To cope with this limitation, Mark et al. propose in [32] a new version of OVMP, called "Evolutionary OVMP (EOVMP)". The proposed approach predicts the cloud users′ demand and optimizes the VMP based on the users′ history. EOVMP is a hybridization of GA, Ant Colony Optimization (ACO), and Particle Swarm Optimization (PSO). It uses the output prediction demand of a demand forecaster. It then allocates VMs using two plans, namely reservation and on-demand. The prediction of the demand forecaster is based on a Simple Kalman Filter (SKF) as the estimation technique, a Double Exponential Smoothing (DES) method to reduce the usage history variations and a Markov Chain (MC) for prediction. The cost obtained by EOVMP is found to be near optimal in comparison to the Mixed Integer Programming (MIP) solution of the deterministic formulation of SIP.

The VMP can be divided into two problems: "Virtual Machine incremental Placement (VMiP)", whereby the VMs can be created, altered, or removed during runtime, and "Virtual Machine consolidated Placement (VMcP)". The dynamic consolidation of VMs in VMcP is an effective way to reduce the energy consumption and to improve physical resource utilization. VMcP, along with ROI, is subject to the work of W. Shi and B. Hong in [31]. Aiming for profit maximization under the SLA and the power budget constraints, VMcP is formulated as a "Multi-level Generalized Assignment Problem (MGAP)". VMs, PMs, and the power budget are considered as tasks, agents, and the resource, respectively. The global manager conducts the placement of VMs at the DC, for a single scheduling period and assuming that this assigned VMs and placement decisions meet the power budget constraints and the SLA requirements. Due to the size of the problem, FFA is also applied, and the results show a low SLA violation rate for the considered experimentation samples.

VII. QoS

In a computation resource-sharing environment, such as cloud computing, QoS would significantly affect the overall performance of cloud services if the placement and migrations of VMs are not efficiently carried out [134], mainly when unexpected network latency or congestion occurs [99]. For instance, one of the major issues encountered in data center is the underutilization of many PMs, while others contain VMs that receive a heavy traffic load, leading certain areas of the network to be congested and to suffer performance degradation. An efficient QoS-aware VMP approach would reduce considerably the traffic transmission across the entire data center, and consequently the congestion and data transfer time. The solutions, introduced in this section, propose VMP algorithms to enhance the QoS, namely the latency, overhead, congestion, data transfer time, and delay.

A. Offline QoS-aware VMP

In [81], Ilkechi et al. address the problem of QoS-aware offline VMP. With the objective of improving the total value

TABLE V: Analysis summary of the relevant/recent work on cost-aware VMP solutions.

Ref	Objectives	Constraints	Algorithm(s)/ Approach(es)/ Policy(ies)	Advantages	Disadvantages	Suggestions to enhance the proposed solutions
[16]	Tradeoffs between business-level SLAs of the hosted applications, the cost of operating the required resources, and the Number of active PMs	CPU, RAM, and total available VMs	Constraint Satisfaction Problem with Choco Solver	The way the conflicting objectives are handled in the proposed CSP solution makes it easier to achieve optimal solutions	Since CSP explores all possible solutions for a set of input data, it cannot be applied to very large datasets	Application of constraint propagation to reduce the number of possible values of each decision variable
[29]	Revenue for the provider and number of successfully placed applications	VM capacity in terms of CPU and memory	Direct Integer Programming	The solution is efficient in scenarios where the VM placement solution must favor gain in computation time rather than precision in the choice of placed applications	- The placement decisions are made to the detriment of QoS - The application sizing is ignored and not considered	Consider QoS requirements and find a fair tradeoff between the gained computation time and the defined constraints
[32]	The costs to the cloud user	Providers resource availability	Stochastic Integer Programming	The solution achieves cost values that are close to the optimal solution and with a faster convergence	Uncertainty about the applicability of this solution in real cloud environments as the solution is based on users' usage history. The solution is vulnerable in regards to the change in the usage pattern	Induce large random fluctuation in the usage pattern and expend the experimentation setup to see how the solution would converge.

of a satisfaction metric related to the overall congestion that reflects the performance of a VMP, two offline algorithms, namely a greedy algorithm and a heuristic-based algorithm, are proposed. The two algorithms find near-optimal solutions regarding the flow demand and communication pattern of the placed VMs. They achieve better results in terms of mean congestion satisfaction and the percentages of link congestions.

B. Online QoS-aware VMP

To improve the scalability of DC networks with traffic-aware VMP (TVMP) when multiple end-users request VMs, in addition to the offline VMP, the case of online VMP is also considered in [23]. TVMP belongs to the class of Quadratic Assignment Problem (QAP), which is considered among the hardest NP-complete problems. In [23], Meng et al. propose a heuristic algorithm to solve TVMP. The algorithm follows a two-tier divide and conquer approach, as it first partitions VMs and organizes them into separate clusters, and then assigns them to hosts at the cluster and individual levels. With the aim of minimizing the typical cost and aggregate traffic, the heuristic algorithm reduces significantly also the computational time compared to the Local Optimal Pairwise Interchange (LOPI) and the Simulated Annealing (SA) algorithms.

To deal with the online QoS-aware VMP, Piao et al. propose in [22] an approach to place and migrate VMs with the objectives of minimizing the data transfer time consumption, to optimize the overall application performance and with respect to some SLA parameters, such as the time requested by the end user. A policy is implemented and tested using Cloudsim 2.0. The proposed policy is compared against the VMP policy adopted by the simulator. The results show that it improves the task completion time, but since the time requirement is not always respected, the enforcement of SLA requirements cannot be guaranteed.

The QoS requirements stated in the SLAs and resource exploitation costs are subject to the reactive and proactive heuristic policies proposed by Cardellini et al. in [27]. The optimal VM allocation is formulated as an MIP. The policies are compared by bargaining computational complexity with system efficiency. Though the obtained SLA satisfaction factor and allocations costs are good, the optimality of the policies solution depends on the fluctuations of the setting parameters. In [64], Wang et al. propose a three-tier algorithm which takes into consideration the energy efficiency and QoS. The first step is hop reduction whereby the VMs are partitioned to reduce traffic transmission. The second step is energy saving

TABLE VI: Analysis summary of the relevant/recent work on QoS-aware VMP solutions.

Ref	Objectives	Constraints	Algorithm(s)/ Approach(es)/ Policy(ies)	Advantages	Disadvantages	Suggestions to enhance the proposed solutions
[20]	Congestion and number of active PMs	Available resources	- A solution based on Linear programming. - A solution based on quadratic programming.	The quadratic programming solution give near-optimal solutions for the dataset considered	The dynamic placements are not handled to meet efficiently the typical workload of modern applications	Study of the applicability of such solutions in different workloads of applications
[22]	Data transfer time	Data access time required by the user.	Constraint Satisfaction Problem	The CSP explores all possible solutions; for environment of small scale, the approach can achieve very good results	- Only 3 hosts, and 3 files not exceeding 4 GB are considered in the experiments - Very costly in execution time	Application of constraint propagation to reduce the number of possible values of each decision variable
[23]	Aggregate Traffic	Operating cost	Approximation algorithm Cluster-and-Cut	1024 VMs and several topologies are considered (Tree, VL2, Fat-tree, BCube)	The benefit of the approach is minimal for an architecture with network load balancing techniques (i.e. VL2)	As also stated by the authors: It would benefit more to combine the optimization objectives considered with server resources objectives (i.e. power, CPU, etc)
[29]	Overhead and number of successfully placed applications	Resource capacity	Direct Integer Programming	For small resource pools, the solver is able to find good quality solutions.	Unable to find feasible solutions for large resource pools	Adapt the model for large resource pools
[81]	Congestion	- Each VM is assigned to exactly one PM - resources utilization	Greedy and heuristic based approaches	Good results in terms of mean congestion satisfaction and percentages of link congestion.	Tested and meant only for single cloud environments	Extend to multiple clouds environments

whereby the maximum number of active servers violating the SLA requirements is defined. Finally, an OpenFlow controller defines the paths that avoid congestion and enable load balancing across the network. Based on conducted experiments, the performance results show that the proposed algorithm saves considerable energy, and both the delay and system throughput are enhanced in comparison with other existing VMP policies.

As a continuation of the work done for the VMP solutions, based on 2-approximation algorithms to minimize the VMs maximum access latency [35], [40], [52], Kuo et al. propose in [67] a new 3-approximation algorithm. More precisely, the problem considers the VM Placement for Data Nodes (VMPDN) with the objective of reducing the maximum access latency between DNs. Each computation node has several available VMs, and the authors considered that to process the stored data, each given DN requires only a single VM. VMPDN is formulated as an MIP. The 3-approximation algorithm designed to solve VMPDN uses the linear programming rounding and the bipartite graph construction. Using the Tree, VL2, Fat-Tree, and BCube network architectures, this solution is compared to the optimal 2-approximation of VMPDN considering a high time complexity. Although it exhibits a worse approximation factor, the 3-approximation algorithm achieves better results regarding maximum access latency values.

VIII. Resource usage

The scheduling, management and optimization of virtual resources are highly important for the performance of VMs. In this section, all the approaches discussed are online-based; some of them are enhancements to previously-introduced offline VMP algorithms dedicated to resource usage optimization.

In [46], Li et al. investigated how to jointly improve the resource utilization, the cost, and the performance of DCs. They accordingly proposed a solution for the online VMP, dubbed "EAGLE". EAGLE design is guided by a multi-dimensional space partition. The model quantitatively defines a resource leak, judging the suitability of resource utilization for the VMP. This judgment is based on a D-dimensional space partition consisting of three domains: acceptance domain, safety domain, and forbidden domain. EAGLE selects the needed PMs to deploy each new VM instance aiming at enhancing the multi-dimensional resource usage and energy consumption by reducing the number of powered-ON PMs. The conducted experimentations, for single and multiple VM requests using several real traces, show that the resource management mechanism of EAGLE saves more energy in comparison to FFA.

In [36], Dias et al. propose an online VMP algorithm to allocate and relocate VMs based on the analysis of usage pat-

TABLE VII: Analysis summary of the relevant/recent work on resource usage-aware VMP solutions.

Ref	Objectives	Constraints	Algorithm(s)/ Approach(es)/ Policy(ies)	Advantages	Disadvantages	Suggestions to enhance the proposed solutions
[36]	Bandwidth usage	CPU and memory	Clustering plus bin packing	- The throughput of the network is considerably improved thanks to moving the traffic from the core switches to the edges switches - Scalability of the solution to be applied in big data centers with very large numbers of machines	- The migration process and how it would affect the complexity of the proposed solution is not studied - The time for the nodes to gather data necessary for moving the resources to the edge is neglected	- As pointed out by the authors, the solution must be improved to consider different data centers models and diverse applications types. - The migration time and influence on the performance of the solution must be studied as well.
[46]	CPU utilization and number of active VMs	Availability	Multi-dimensional space partition	The resource fragments used are improved considerably in the proposed solution.	The solution's local optimization performance impacts the overall resource utilization balancing	The constraints defined in the solution don't consider the case of multi-tenant cloud environments
[51]	Resource wastage (CPU and memory) and power consumption	PMs capacity and availability	Ant Colony Optimization	- The first application of Ant Colony Optimization in VMP - The solution is suitable for large size of data centers with thousands of VMs.	N/A	The constraints defined in the solution don't consider the case of multi-tenant cloud environments
[95]	Resource wastage and power consumption	PMs capacity	Biogeography-based optimization	Approach converges to the optimal solution	Limited data in the experimentation	Study the performance of the solution in larger data centers with different types of applications

terns of CPU, traffic, and memory. The patterns are extracted based on the exchange of a high amount of data among VMs. Relying on graph theory, the correlated VMs are aggregated and allocated to servers chosen based on the distance to each other such that the traffic congestion is reduced. With the goal of achieving minimum traffic congestion, a solution is proposed as the combination of a modified Girvan-Newman algorithm and allocation scheme specifications. The conducted experimentation showed that the proposed VMP approach improved considerably the traffic distribution of the core traffic. The results also showed a feasible execution time and an improvement of the network traffic quality compared to "no-management".

Along with the ongoing advances in virtualization technology, servers can be sliced into multiple execution environments. Those isolated environments are deployed on VMs. It becomes challenging to satisfy the received tasks and requests, and manage the available virtual resources. Based on a two-level control approach meant for automating virtual resource management, Xu et al. in [13] expand this offline approach to a new global controller at a virtualized DC level [19]. This controller defines the resource allocation of a VM and answers to the requirement of a shared hosting environment on the virtualized platform infrastructure for the applications of end users. This controller is based on an improved GA combined with a fuzzy algorithm. As objectives, it aims at minimizing the resource wastage, the cost of thermal consumption, and power consumption. Compared to bin packing algorithms, the solution makes better usage of the available multidimensional resources by reducing at the same time the energy consumption.

A novel solution, introduced by Zhenga et al. in [95] and called "VMPMBBO", considers a VMcP system based on a resource wastage model and a power consumption model. This solution uses "biogeography-based optimization (BBO)", which is known to converge to optimal solutions, in order to optimize the VMP with the objectives of reduced power consumption and resource wastage, as well as to balance the server loads and storage among VMs. Although servers are assumed to be homogeneous, and the VM deployment requests consist of pairs of CPU and memory demands, the extensive simulations show that the solution achieves better convergence and outperforms the "Multi-objective Grouping Genetic Algorithm (MGGA)" and the offline VMP solution "Virtual Machine Placement Ant Colony System (VMPACS)" [51].

IX. Reliability-aware VMP

Minimizing the number of involved hosts should also consider prevention of unforeseeable hardware failures which may raise the need to ensure a satisfying level of reliability for VMs and the services they provide. Relying on redundant configurations using VMs can be an effective countermeasure. To do so, it is obvious that the online VMP would be more of interest when considering reliability issues, as the chances of

TABLE VIII: Analysis summary of the relevant/recent work on reliability-aware VMP solutions.

Ref	Objectives	Constraints	Algorithm(s)/ Approach(es)/ Policy(ies)	Advantages	Disadvantages	Suggestions to enhance the proposed solutions
[21]	Reliability	Available virtual machines and number of hosting machines	Multiple k-redundancy method	- Among the first to consider reliability in virtualized environments. - The consolidated servers are believed to be more reliable and with a low cost.	Despite the fact that the solution was intended for distributed hosting servers, the environment in which the method has been tested consisted on only one server handling web applications' HTTP requests	Extend the approach to multiple distributed hosting servers
[24]	Reliability and L2 cache misses	Available cores	Four placement strategies, S4P1, S4P2, S4P3, and S4	- A VMP solution dealing with the objective of L2 cache misses was proposed for the first time in this paper. - The obtained results reveal that enabling VMs to run on any available core, rather than the main one, noticeably enhances the miss rate of L2 cache.	The size of the experimentation environment.	Adapt the approach to support a large number of hosting servers and cores.
[100]	Reliability	Network resources	Recursive heuristic-based algorithm	- Only online reliability-aware VMP solution found in the literature. - The experimentation results show that the solution improves the reliability and reduces in the same time the network resources usage.	The applicability of this costly approach to a larger experimentation setup.	N/A

facing unforeseeable failures are higher. Also, it is worth noting that redundant configurations imply an important increase in resources utilization and it must take into consideration the additional QoS issues to likely encounter, mainly regarding network congestion and aggregate traffic, and mostly in online VMP scenarios. Having said that, it is worth noting that most of the work mentioned in the literature is dedicated to offline VMP, except in [100].

Machida et al. in [21] present a VMP method that establishes a redundant configuration against host server failures with fewer host machines. In consolidated server systems with various hosted online applications, a redundant configuration of VMs is made in anticipation of host server failures. This minimum configuration is meant to achieve k-resiliency for VMs. The k-resiliency means that there must be a possibility to relocate a VM (without affecting other VMs) to a non-failed host as long as there are up to k host failures. The problem is defined as a combinatorial optimization problem. The solution obtains a redundant VMP based on the multiple k-redundancy method, which leads to a theoretical minimum number of host machines. The obtained hosting machines have lower cost and higher reliability.

In [33], Bin et al. model the k-resiliency conditions as input constraints to a Generic Constraint Programming (CP) solver with the objectives of achieving high availability and respect constraints such as the resource feasibility. The proposed technique is based on two fundamental points. The first one consists in merging "Hardware Predicted Failure Analysis alerts (HwPFA)" and live migration to support smooth operations of active VMs. The second one relates to the fact that resiliency can be achieved by the creation of a transformed VMP that includes shadow VMs. The results show that a load balancing optimization is obtained with a satisfying k-resiliency.

The k-fault tolerance is also subject to the work of Zhou et al. in [100]. To enhance the reliability of server-based cloud services, a network-topology aware redundant VMP solution is proposed to minimize the consumption of network resources, under the k-fault tolerance constraints, in the case of VM failure recovery using backup VMs. The proposed approach first relies on a host selection process. A Point of Delivery (PoD) in the DC with enough resources is selected, and the residual capacities are provisioned for later usage, then an optimal redundant VMP is carried out using a recursive heuristic-based algorithm. Finally, a recovery strategy decision is triggered, where each VM in the failure state is mapped to a backup host. This mapping problem is formulated as a "maximum weight matching in bipartite graphs problem".

Based on the characteristics and nature of the DC network, a recursive heuristic-based solution selects appropriate hosts and determines the needed optimal placements. The experimentation results show that the solution improves the reliability and reduces at the same time the network resources usage.

With the same concern to guarantee the reliability, Chen et al. in [94] proposed a new scheme based on an adaptive selection of fault-tolerant strategy dubbed "SelfAdaptionFTPlace". SelfAdaptionFTPlace is carried out in three stages:

- The constraints are extracted from the application requirements.
- With respect to the defined constraints, fault-tolerant strategies are selected.
- The VMs are placed based on the defined strategies.

The constraint model takes into account $i)$ the cloud resources consumption of an end user, $ii)$ the response time (i.e. the time needed for a given application request to receive a response from the cloud for a given end user) and, $iii)$ the maximum failure rate tolerated by the end user for a given application. In the first phase of SelfAdaptionFTPlace, the best evaluation function value of a VMP is obtained based on the constraint factors. In the second phase, based on the output of the first phase, the placement decision is made. The performance evaluation demonstrates that SelfAdaptionFTPlace obtains better response times, failure rates and memory usage compared to some previously proposed methods, such as RandomFTPlace [44], NOFTPlace [21] and ResourceFTPlace [26].

Parallel to placing redundant VMs, placing VMs on multi-core processors in caches, rather than the default placement schemes, enhances performance considerably. This motivated the work of Emeneker et al. in [24]. The authors used Oprofile and Xenoprofile for gathering cache miss data to test the performance of multi-core cache structure on applications running inside Xen VMs. The results of the benchmark of several placement strategies are applied in the cases of placing a single VM and two VMs. In both cases, the VMP schemes are evaluated under several system specifications (e.g., quad core system, P2, S2P1, and S2P4). The obtained results reveal that enabling VMs to run on any available core, rather than the main one, noticeably enhances the miss rate of the L2 cache.

X. Load balance-aware VMP

The work, presented by Hyser et al. in [12], introduces an autonomic controller, which monitors the activity of VMs. Using advanced policies, it achieves a dynamic workload placement. In addition to less frequent overload situations (load balancing), the controller reacts, in accordance with DC policies, to the variations in physical hosts utilization and VM loads. It also proposes some components to improve the cooling loads and power consumption.

For a semi-homogeneous DC configuration and when the usage is quite frequent, Li et al. studied the benefit of encouraging multi-tenancy in DCs [49]. They propose a load balance oriented VMP scheme that hierarchically and vertically (top-down layers) places VMs with the objectives of maximizing both the machine and bandwidth elasticity, thus, minimizing the utilization of PMs and link resources. Two simulations were conducted for DCs with heterogeneous and semi-homogeneous configurations. Using a three-layer binary tree structure topology, the simulations show the performance of the scheme against the solution produced by a brute-force search regarding VM number, cluster utilization, and link capacities.

The objectives of profit maximization, load balancing, and resource wastage minimization define the work of Adamuthe et al. in [50]. These authors proposed a new model for the scheduling of virtual resources in the cloud using three GAs with pre-defined objectives. The performance of each solution is evaluated based on the cost incurred due to constraint violations. Two GAs, namely the baseline GA and "Non-dominated Sorting Genetic Algorithm (NSGA)", defined duplicate solutions and suffered from a premature convergence. To cope with this limitation, a new version of NSGA; NSGA-II, is produced, handling the VMP problem as a minimization problem. A comparison with the results of GA, with the same objectives of reducing power consumption and resource wastage, shows the efficiency of the multi-objective ant colony system algorithm proposed by Gao et al. in [51]. The solution algorithm is tested using the same server node and VM dimensions, namely CPU, and memory, but just for the case of static placement (offline).

XI. Virtualized Network Functions placement

The placement of virtual mobile core network functions is addressed widely in the recent literature. Since the optimal placement of VNFs is known to be NP-hard [2], several strategies are proposed, and many issues related to the VNF placement arise. As depicted in Fig. 6, the VNF placement can be classified into two main categories:

- The general placement: the focus here is to define efficient placement strategies and policies based on chains, replications, forwarding graphs, etc.
- The placement of specific network functions, such as Packet Data Network Gateways (P-GWs), Serving Gateways (S-GWs), and transcoders.

This classification is not only motivated by the fact that there is a difference between the several use cases addressed in the first category (see Section III), but also because the several solutions, presented in the second category, aim to enhance specific metrics that are related to specific network functions (e.g., minimizing S-GW relocations and reducing the cost of the path to P-GWs).

A. General Virtualized Network Functions placement

The VNF placement model proposed by Moens et al. in [75] considers the management of both service and VM requests in a non-restrictive network topology. It handles the two request types differently and is evaluated for two types of service chains, through a scenario of a small service provider. Based on Integer Linear Programming (ILP), the proposed algorithm finishes in few seconds (i.e., 16 seconds) which makes it quick to cope with sudden changes in demand for resources, which could be due to NFV burstiness. In this solution, the virtualized

TABLE IX: Analysis summary of the relevant/recent work on load balance-aware VMP solutions.

Ref	Objectives	Constraints	Algorithm(s)/ Approach(es)/ Policy(ies)	Advantages	Disadvantages	Suggestions to enhance the proposed solutions
[12]	Load balancing the CPU, LAN, and Storage	Availability	- Load Balance Policy based on Simulated Annealing algorithm	The solution provides several load balancing objectives (CPU, LAN, disk)	-The policy is applied only for four servers - It is a centralized-controller based solution - High complexity of the Load Balancing Policy to be able to balance all the hosts' loads which is costly in matter of computation and implementation in very large topologies	- The performances could be enhanced by using ACO or GA - Authors could consider a distributed solution based on separate controllers
[49]	Load balance and resource usage (bandwith and link usage)	Link capacity	Three-layer binary tree	For a small scale cloud environment with both semi-homogeneous and heterogeneous datacenter configurations the proposed approach gives optimal solutions	Tested only on a small scale environment	Study of the applicability of such an approach in large multi-tenant cloud environments
[50]	Load balancing and resource usage (CPU and memory)	PMs capacity	GA, NSGA and NSGA-II	For a small scale cloud environment the proposed approaches give good quality solutions	- Problem of duplicate solutions and premature convergence in some of the proposed approaches - Size of the experimentation setup	Study of the applicability of these approaches in datasets with very large number of application requests

services handle the spillover and the hardware handles the base load [96], [97]. These restrictions are discussed in several works.

With the objectives of minimizing the usage cost of link and node resources, Baumgartner et al. addressed in [91] the placement of different VNFs, such as S-GW, P-GW, Home Subscriber System (HSS) and Mobility Management Entity (MME), excluding VNFs of the RAN. They also considered the VNF requirements (i.e., processing, storage, and bandwidth) excluding latency on the end-to-end network and that on VNF nodes. The RAN domain, including the firewall, load balancing, and virtual nodes are addressed in [92]. In this work, Riggio et al. aim at satisfying the VNF requirements (i.e., memory, CPU, radio, storage and bandwidth), while minimizing the cost of mapping VNFs and that is without taking into account the end-to-end latency. This metric, being of vital importance for edge cloud, was considered in [112], in addition to other QoS requirements, such as the response time and the real-time requirements. Represented as a Multiple Objective Decision Making (MODM), the main objectives are to improve resource utilization, reduce overload, and answer to the SLA constraints.

In [89], Adis et al. consider the problem of "VNF placement and Routing Optimization(VNF-PR)", for two types of forwarding latency regimes, with respect to constraints of compression and decompression, and under both Traffic Engineering and NFV objectives. The designed scheme handles, in a relatively short execution time, large experiment instances of the problem and takes into consideration NFV deployment strategies based on realistic settings.

Applying NFV to the Evolved Packet Core (EPC) raises the need for optimal network function placement. This raises challenges related to the consequent delay budgets among cellular core components, and the management of communication among data and control plane elements should be well handled. In this vein, Yousaf et al. presented in [61] the concept of a Soft Evolved Packet Core (softEPC), which distinguishes the typical EPC functions from mission-oriented services and specific hardware by instantiating them in a decentralized, load-aware and on-demand manner. The gained performance from using softEPC is analyzed and proved to be an enabler of a dynamic placement of mobile network functions, improving load-balance, bandwidth, and link utilization. In [61], control plane entities (e.g., Policy and Charging Rules Function - PCRF and MME) and some deployment-related parameters (e.g., latency and mobility) are not considered. Some of these requirements, namely latency, memory, and CPU are addressed by Oechsner et al. in [90]. Indeed, as a continuation of the work done in [21], [23], [28], the authors in [90] describe a practical solution and case study for placing a network function in an OpenStack-based cloud environment. It is meant to serve as a practice-oriented scheme for placing virtualized functions in infrastructures.

The proposed solution is split into two parts:

- "Structural Aware Planner (SAP)": SAP takes as inputs the application and DC description. Then, by considering the constraints of availability and connection, it builds tree models.

- "Demand Aware Planner (DAP)": DAP arranges VMs in groups, places the groups in clusters (per server) and checks if the requirements of each VM are satisfied.

Along with the VM requirements of the network functions, the cost saving is also considered in the oriented optimal placement scheme proposed by Yousaf et al. in [85]. The VNF placement is treated by analyzing the cost incurred by two constraint-based deployments, namely "Vertical Serial Deployment (VSD)" and "Horizontal Serial Deployment (HSD)". These strategies enable an initial deployment of VNFs, considering a virtualized mobile network infrastructure and providing an Evolved Packet Core As A Service (EPCAAS) which respects the functional and administrative constraints. The cost of VNF placement can be reduced using algorithms such as Bin Packing, Simulated Annealing, Ant Colony, Transient cooling effects, N-dimensional set and so on for VM placement within the same DC [42], [86]. In the same vein, Bagaa et al. propose a complete Core Network as a Service in [158], [166] over a federated cloud, deploying virtual instances of key core network functions, namely MME, SGW, PGW, the Access and Mobility Management Function (AMF), Session Management Function (SMF), Authentication Server Function (AUSF), and User Plane Functions (UPFs). Their solution includes an efficient coalition-formation game-based VNFP algorithm which finds an optimal tradeoff of QoS while reducing the deployment cost after deriving, based on MIP, the optimal number of virtual instances to meet the requirements of specific mobile traffic.

B. VNF placement and the VNF forwarding graph

The VNF-FG design is proved to be an important part of the VNF placement problem. Mechtri et al. propose in [116] an analytically-based approach as a solution to the problem. The proposed approach is an Eigendecomposition extension. Eigendecomposition is the factorization of a diagonalizable matrix, represented in terms of eigenvectors and eigenvalues, into a canonical form. The link mapping, using the extended Eigendecomposition of the request, is faster, more scalable and improves the resource usage when applied for several use cases and metrics (e.g., the system load and network connectivity). Furthermore, the model relying on the new analytic Eigen-decomposition approach achieves better consolidation results compared to other schemes.

In [122], Cao et al. propose a new method based on flow design and service requests for generating VNF-FGs. Based on the generated VNF-FGs, the NFV environment is modified with additional mapping modes and physical nodes, enabling the VNFs mapping. Two genetic algorithms are tested within this framework, the "Multiple Objective Genetic Algorithm (MOGA)" and an improved NSGA-II. The experimentation demonstrates that the improved NSGA-II and the VNF-FG design reduce considerably the total bandwidth consumption.

C. VNF placement and the VNF Chain Placement Problem

The VNF Chain Placement Problem (VNF-CPP) is another VNF placement-related problem, which is known to be NP-Hard. It is important to find placement schemes that can scale with the size of the problem and find good quality solutions [133]. Moens et al. were the first to address VNF-CPP in [75] by formalizing it as an optimization problem.

In [117], an ILP-based model is proposed by Sun et al. to minimize the total deployment cost and increase the service providers′ profits by predicting the VNF requirements. The proposed solution can also reduce the probability of a service chain request being blocked. However, the ILP model has limited applicability and is especially efficient in cases of small numbers of user nodes. Bhamare et al. propose in [126] a novel Affinity-Based Approach (ABA) to cope with this limitation. The approach considers different user-levels with different user delay tolerances satisfying QoS as well as SLA requirements. Also, the traffic-affinity between VNFs is taken into consideration for the placement in the cloud. A performance comparison between the proposed ABA heuristic and a Simple Greedy Approach (SBA) using the First-Fit Decreasing method (FFD) shows the quality of ABA with only a marginal increase in execution time.

In [125], Luizelli et al. incorporate a Variable Neighborhood Search (VNS) meta-heuristic, for efficiently exploring the placement and chaining solution space. VNS aims to minimize the required resource allocation while meeting the network flow requirements and constraints. The algorithm can find efficiently feasible and high-quality solutions in scenarios that are scaling to hundreds of VNFs. In [123], a service chain is created, consisting of a number of VNFs which facilitates a specific use case for many users and follows a multi-tenancy single-feature approach. A hierarchical architectural framework is proposed for the VNF placement, following the general guidelines of major standardization communities (e.g. ETSI) that leverage the capabilities of SDN and cloud technologies, and which is proved to be highly complementary to the NFV paradigm. Based on MIP, four heuristic algorithms, namely baseline, consolidation, load balancing, and worst performance are proposed to cover a large range of complexity and performance levels, such as the number of created cloud nodes, CPU utilization, cost and link utilization. The link utilization, while guaranteeing resiliency for failures in single-node, single-link, and single-node/link, was subject to the work of Hmaity et al. in [132]. They focused on the case of latency sensitive services, and considered the underlying routing and capacity constraints. The experimentation results showed an interesting decrease in virtual nodes needed with a fair tradeoff between node capacity and latency of the deployed service chains. Unfortunately, to assess the efficiency of the proposed heuristic model, larger instances should be used.

In [124], Khebbache et al. proposed an optimization method based on a multi-stage graph to improve scalability and cost. The algorithm is compared against an exact 2-matching method. Experimentation on complex and longer chains proved the scalability and efficiency of the proposed method as well as its ability to find sub-optimal and good quality solutions. The performance assessment is based on the following metrics:

- The convergence time: defining the time needed by the algorithms to find a sub-optimal or optimal solution.

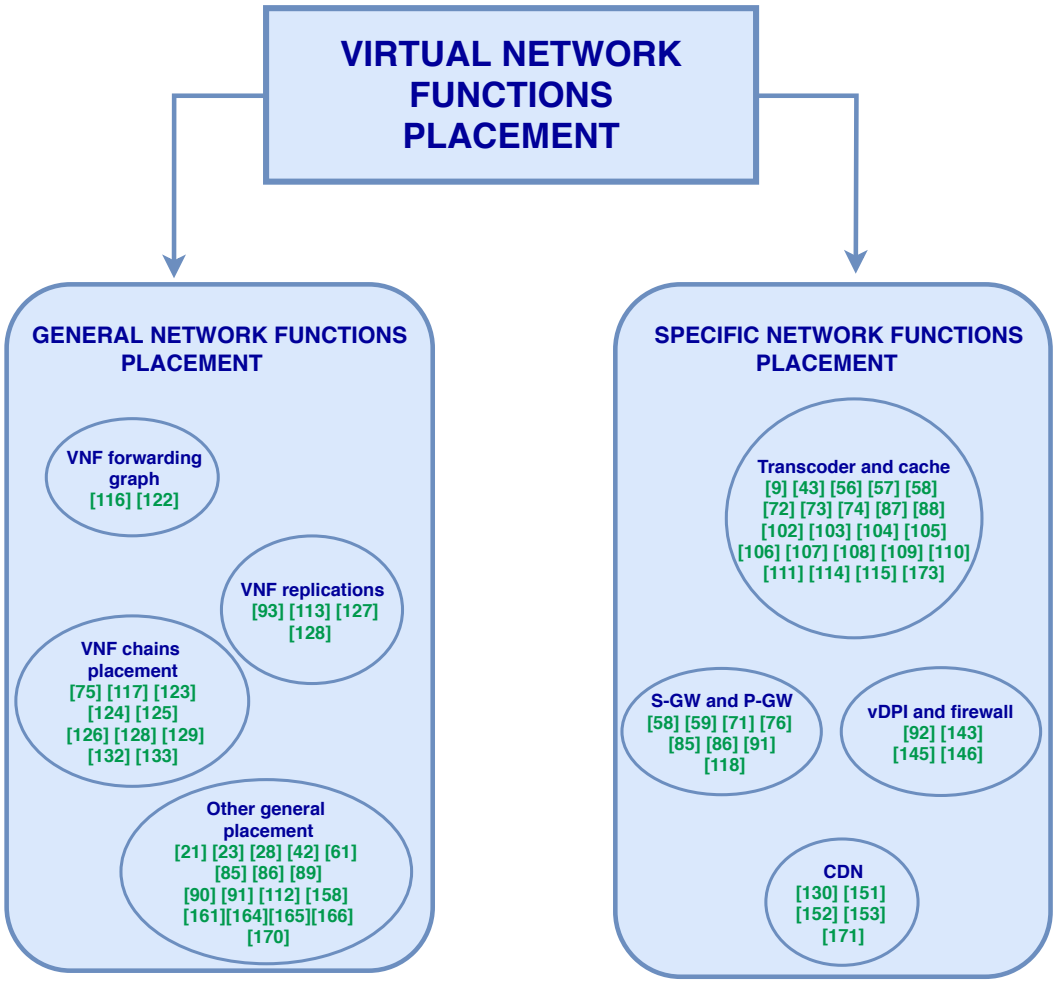


Fig. 6: VNF placement classification.

- The acceptance ratio: indicating the average number of VNF-FG requests accepted for being hosted in the physical infrastructure.
- The average cost: being the sum of reserved resources in the infrastructure. The considered resources are the processing capabilities of the servers and the available bandwidths on the links.
- The Average cost/revenue ratio.

In [129], Dietrich et al. considered the deployment of the main components of EPC as VNFs in DCs close to base stations (i.e., edge cloud), ensuring elasticity in resource provisioning and better load balancing. They introduced a request model and network model for the cellular core network, expressing sequences of EPC VNFs as service chains and proposing a linear programming formulation for the computation of VNF placement aiming to balance optimality and time complexity. Using a realistic evaluation environment and CPLEX for the linear programming models, the performance results show that the linear programming model achieves better load balancing, higher request acceptance rate, and better resource utilization compared to the greedy algorithm, widely used as a baseline.

D. VNF Placement and VNF replications

VNF replications have been studied closely as the traffic directed to DCs has a significant impact on network load balancing. The impact is even more significant when this traffic has to traverse an ordered sequence of VNFs (sub-section XI-C). With the virtualization environment, VNF replication is now made possible [93]. On this matter, Carpio et al. studied in [113] the problem of VNF placement using replications. Knowing that VNF replications help to balance the network load, they designed and compared three optimization methods. A linear programming model is used for small networks, while for large networks, GA and the Random Fit Placement Algorithm (RFPA) are used to decrease the computation time for the allocation and replication of VNFs. Another work by the same authors in [127] proposes a new linear programming based model, but this time to find an optimum placement of functions aiming at a tradeoff between the minimization of two objectives, namely the link utilization and CPU resource usage. The results show how the model balances the utilization of all links in the network using minimum resources.

A completely different approach, using replications, is proposed by Pham et al. in [128]. First, a “Sampling-Based

Markov Approximation approach (MA)" is proposed to the VNF placement problem. This method needs a long time to find a near-optimal solution which makes it unpractical. To cope with this issue, the Matching Theory is combined with MA and is found to reduce the total cost, achieving a reasonable execution time compared to the existing approaches. To simplify the complexity of service chains within this solution, VNF replications are added, and the results show that this helped to reduce the traffic cost and the number of activated nodes.

XII. Specific network functions placement

A. Transcoder and cache placement

Today, video streaming services are omnipresent. Users are seeking faster service delivery and expect higher quality videos. As the connection speed and streaming system abilities have an influence on the streaming capabilities, the challenge is to find a fair tradeoff [109]. Furthermore, many other variables could be game changers, such as the device type, the screen size, CPU, GPU, the available bandwidth, network traffic, distance from the server hosting the video, the type and version of browser, and used plug-ins such as Flash or Silverlight.

Different methods can be used to transcode videos, and the ability to move transcoding resources could help in optimizing several parameters (e.g., bandwidth and latency of the network), ultimately ensuring better Quality of Experience (QoE) [110], [111], [173]. In [56], Farrow et. al. address the transcoding resources optimization problem using a heuristic algorithm design. This design takes into consideration the constraints of computational and network requirements. The solution is shown to achieve better computational resource usage. However, it shall be highlighted that the authors do not discuss in details the dynamic movement mechanisms of transcoders.

The dynamic migration to different locations, while streaming is taking place, is discussed in [87], along with optimal placement in the network. OpenFlow is used to optimize the transcoder migration during streaming, and a heuristic is provided to solve the transcoder placement optimization problem while achieving a similar result to that of the GA. It provides the capability to optimize the transcoder placement using the placement algorithm multiple times during a transmission, providing a highly optimized system throughout the duration of the transmission, even with a client population shift. This principle can be adapted for use with other scenarios, such as migrating transcoders based on reducing DC costs as well as for the sake of energy saving.

The throughput can be improved by more than six times, using only one-seventh the number of processor cores, when the NFV (rather than a standard server) orchestrates the accelerated video transcoding. In [58], Basta et al. propose an interesting approach to transcode videos on general-purpose servers with video accelerator add-in cards. This approach does not only provide an overall much lower cost, but deployment and operation cost comes down because the solution can be managed like other servers in the DC or central office. It takes advantage of NFV and the open cloud computing architecture. This gives service providers the flexibility to deploy video transcoding on nearly any server, to add services as easily as a software update, or to run complementary applications, such as billing or QoS, on the same server.

Related to the placement of transcoding resources, content caching has been widely studied. In [88], [102], the authors developed game-theoretic models to evaluate joint caching and pricing strategies among access networks, transit networks, and content providers. The research work presented in [43], [72], [73] focused on content caching in wireless networks, and on exploiting the backhaul links for collaborative caching [74], [103]. Recently, the authors in [104], [105] proposed a hierarchical cooperative caching in a Cloud-RAN (C-RAN) whereby the cloud cache is introduced as a bridging layer between the core-based and edge-based caching schemes. The authors propose an online cache management strategy with less complexity, consisting of proactive and reactive algorithms. The cache distribution is carried out using the first one, and the cache is replaced using the second, to minimize the average delay cost of all content requests. In the same line, Mosleh et al. defined in [106] the problem as a "Mixed Integer Nonlinear Programming (MINLP)", solved by a coordinated data assignment algorithm in C-RAN to enhance the QoS using two defined matrices, one for pre-coding and another for cache placement.

To address the multi-bitrate video streaming, several research works have focused on Scalable Video Coding (SVC) [57], [107], [108]. In [9], [114], the authors consider caching and processing multi-bitrate (multi-version) video streaming, but only on one cache entity, as opposed to the collaborative scheme of multiple caching/processing servers. Also, they propose to deploy collaborative caching in a Multi-Access Edge Computing (MEC) network, whereby the MEC servers can assist each other for both caching and transcoding of multi-bitrate videos. The problem of joint collaborative caching and processing is formulated as an ILP to minimize the total cost of retrieving video content over backhaul links, which is resolved using Joint Collaborative Caching and Processing (JCCP) in [115].

B. S-GW and P-GW placement

Both S-GW and P-GW have key roles in the EPC architecture [58]. In [59], the S-GW placement problem is presented as an NP-hard problem. With the objectives of minimizing the relocation of S-GWs, it is necessary to ensure optimal planning that takes into account the observed mobility of users and their data traffic load and determines an optimal number of S-GW instances that must be created. Therefore, one must find a tradeoff between reducing the data transfer, among UEs and over Service Areas managed by S-GWs, and the reduction of the number of instances created for virtual S-GWs. Basta et al. [76] discuss the virtualization of mobile gateways, namely S-GWs and P-GWs hosted in DCs. They analyze the optimal placement by taking into consideration the load overhead in the transport network, the overhead in the SDN controller and other parameters, such as the potential number of DCs and the delay in the data-plane.

Recently, mobile operators tend to leverage on the NFV and SDN capabilities to deal with the increase in mobile data traffic. In this context, Bagaa et al. proposed in [71] a new scheme to create virtual instances of the P-GW and to effectively place each virtual instance for UEs while ensuring QoE. The first objective is to minimize the operator cost by increasing both the number of P-GW instances and UEs using the same P-GW. Whereas the second objective seeks to minimize the amount of traffic difference between P-GWs (e.g., load balancing). This process is modeled by a nonlinear optimization problem, which is proved to be an NP-hard problem. Therefore, the authors propose three heuristic algorithms to solve it: "Optimal Network Function Placement for Load Balancing Traffic Handling (ONPL)", a "Greedy Algorithm" and a "Repeated Greedy Algorithm (RGA)". The performance results demonstrate that the proposed schemes yield almost optimal performance. In the same way, Yousaf et al. [85] proposed a fine-grained scheme based on the computing Reference Resource Affinity Score (RRAS) values of each hosted VM for the optimal management and decision of VNFs. This approach can optimize the lifecycle management operations on the VNF instances and minimize the number of costly VM management operations. The research work presented in [86] proposes three VNF placement algorithms for a carrier cloud to place P-GWs and S-GWs with the objectives of minimizing the path cost between the gateways (i.e., P-GWs) and end users and optimize their sessions´ mobility with respect to the constraints of 3GPP specifications. The architecture consists of the cloud domain composed by distributed DCs over a geographical area and the RAN domain consisting of access points. The first algorithm "Avoiding S-GW Relocation (A-SGWR)" achieves the defined objective of favoring the S-GW relocation; the second one "Shortening Path Length between eNBs and PDN-GW VNFs (S-PL)" enhances the path between UEs and the respective P-GWs while the third one "Fair and Optimal SGW Relocation and data delivery Delay (FORD)" finds a fair placement with a tradeoff between the given objectives, i.e., S-GW relocation and the delay overheads, based on the Nash bargaining.

With the same objectives, the work in [118] proposes a modeling, using constraint programming, for the placement of both S-GWs and P-GWs, to minimize the number of VNF instances, the number of S-GW relocations and the length of the path between the P-GW and end users. Several types of services, end users requirements and geographically distributed DCs are taken into consideration. A resource controller receives inputs regarding user behavior, service characteristics, and other metrics, and provides as outputs an optimal configuration for S-GWs and P-GWs based on the location of DCs.

C. Virtual Deep Packet Inspection placement

It is true that NFV brought many opportunities to service providers in the revolutionary shift to operate telecommunication networks at low cost and support rapid introduction of new services into the market. However, it has equally brought many security challenges along the way and those are at the cloud platform, the network, and the application levels. The content must be secured and reliable at all these levels [147]–[149], and with a high level of availability [150]. In this vein, this section will discuss the research work dedicated to the specific case of placing virtual DPI (vDPI) and firewall functions.

M. Bouet et al. proposed in [145] a GA-based method to deploy DPI engines in a cost-effective manner. They aim to reduce overall costs, computed based on the following metrics:

- The number of deployed vDPIs.
- The number of flows that were not analyzed: DPI filters packets to examine the corresponding packet flow data, looking for viruses and any other possible inconsistency or threat.
- The network load.

This problem was formulated as an Uncapacitated Facility Location Problem (UFLP) [143]. The GA operations were defined, based on how the initial population is generated, the selection, crossover, and the mutation operations, as well as the fitness value which reflects the defined objectives. Varying the traffic from dense to random, the global cost decreased by more than 58% when relaxing the capacity used per link. Due to the small-scale experimentation inputs, a new formulation of the problem, using ILP and a heuristic implemented in the Java Universal Network/Graph Framework, was proposed by M. Bouet al al. in [146]. The solution was tested against realistic conditions using a large-scale dataset on the high bandwidth backbone GEANT. Despite the fact that the solutions cannot be used in networks of a scale exceeding 35 nodes for the linear programming solution and 300 nodes for the heuristic solution, they are able to find a fair tradeoff between the network footprint induced and the vDPI cost function.

D. Replication of Content Distribution Networks

The virtualization capabilities made it possible to go beyond CDNs that are running on dedicated infrastructures (i.e., physical CDNs) and replace them with the concept of virtualized CDNs which help to reduce the cost of using dedicated servers and third-party content providers. With today's important computing and calculation capabilities (e.g., multiprocessor systems and High Performance Computing) to support multiple VMs, delivering video content using CDN functions can run in one VM on the same physical server where other VMs are operating other services.

Cahill et al. proposed in [151] three different replication-based placement algorithms for CDNs:

- The first algorithm makes CDN replications based on the number of hops it takes to get to the client. Organized as clusters, each clients' cluster receives the requested content from the nearest proxy.
- The second algorithm calculates a cost function based on link and storage utilization. Each proxy evaluates the cost value of each cluster it serves. Based on these values, the algorithm decides if a replication is needed or not.
- The third algorithm is an enhancement of the second one. It adds a delay to the cost calculation procedure triggered when a client joins.

Although the experimentation concerned the case of 100 clients, the last two algorithms were compared to the widely used closest-proxy algorithm and better results were obtained in terms of number of CDN replicas, link cost and storage cost when streaming full high quality movies to clients. Another replication-based placement of CDNs was proposed by Jiang et al. in [152] for the case of hybrid CDN-P2P architecture. It is an enhanced version of the Replica Placement Algorithm (RPA), used by default for CDNs [153], and is based on a heuristic algorithm which finds the optimal set of placement decisions for surrogate servers with the minimal placement cost. The results show that this enhanced version achieves better transport and storage costs, compared to the default RPA.

In [130], Retal et al. proposed a Content Delivery Network as a Service (CDNaaS) platform for the management of a high number of videos deployed on virtualized caches, transcoders, and streamers. On the one hand, the platform offers the possibility for the CDN slice owner to add videos and specify their resolutions. On the other hand, these videos are streamed to the consumers of the CDN slice. The assignment of VM flavors and their adequate locations are based on two ILP solutions. With the objectives of maximizing the QoE of the streaming service while respecting the total cost paid by the user, the experimentation results using the Gurobi Optimization tool prove the efficiency of the proposed solutions. Another CDNaaS platform was introduced by Benkacem et al. in [171] for the dynamic deployment and life-cycle management of virtual CDNs slices in multiple cloud domains. They proposed mechanisms for allocating the needed VNFs for each CDN slice based on two ILPs formulations and solved based on the bargaining game theory with the objectives of minimizing the cost and maximizing the QoE. The experimentation results show the effectiveness of the framework to find an optimal tradeoff solution between the cost efficiency and QoE.

XIII. Key challenges and lessons learned

Admittedly, NFV is still in its early stages. To ensure the ultra-short latency, high QoS, service reliability and security promised in 5G, many key challenges still need to be thoroughly addressed. For the work assessed in this survey on VMP, as it could be seen in Tables IV to IX, it is clear that there are still many unresolved issues, mainly with regard to the size of cloud setups and multi-tenancy considerations. Also, there is still room for work, particularly, concerning service reliability and k-resiliency, and the optimization of overhead and data transfer time in case of online VMP.

Concerning the VNFP, based on the different solutions discussed in this article, and other solutions offered in the virtualization era as alternatives to the traditional existing infrastructure, most providers are still learning about the challenges that arise from common infrastructures, typically in terms of complexity. The management of service quality, dependencies, performance, and scalability becomes extremely difficult within the highly dynamic NFV ecosystem. The research work dedicated to the placement of network functions is encouraging but still many issues still remain unresolved. Fair tradeoffs between deployment related parameters, link utilization, cost for both service providers, and end users are lacking. For instance, in [21], [23], [28], [61], [92], many deployment parameters such as latency and mobility are not considered, while solutions which consider such parameters as in [93], [113], [117], [125], [128], [129] seem to have limited applicability and are more efficient in cases of small numbers of user nodes, due to their complexity costs.

The repackaging of network functions as virtual appliances must fulfill the promise of NFV to offer agility and cost reduction (i.e., reduced CAPEX and OPEX). Different stakeholders must look toward leveraging automation of processes and orchestration to serve these objectives. It is of vital importance to validate physical and virtualized network functions and infrastructures to do benchmarks and ensure that the capacity and performance requirements are met. This process should take into account a complete testing of NFV infrastructure (NFVi) and physical network functions. Also, more software should be written in a "cloud native" manner with a deep embedding of the cloud infrastructure.

Several lab-based simulations lack realistic data and the means to mimic the important workload, dependencies, and conditions which could permit to propose efficient policies and solutions for virtual resources' provisioning, placement, recovery and maintenance. Verizon, in collaboration with Red Hat and Big Switch, is one interesting success story worth mentioning which successfully was able to deploy large-scale network functions [7]. However, the director of NFV planning at Verizon mentioned that they had to face many challenges with "one size does not fit all for all NFV workload connectivity" being the main issue, along with the need for high availability, SSL support, IPv6 support, scale testing, and building in the needed capabilities [7].

We believe that the focus should be on these critical operational requirements. The best practices and the agile tools in hand, such as artificial intelligence, service modeling, and prediction must be used to improve the allocation of resources (i.e., VNFs and VMs) in a standardized approach. The pillar is to enable efficient simulation tools (if not real tests such as Verizon's) that could reflect the UE consumption of services, the nature of the heterogeneous infrastructure and the SLA requirements to fulfill. Also, we observed that many VNF use cases have not been addressed yet (if partially) by the research community, namely the home and business gateway virtualization (i.e., vCPE, SD-WAN), Virtual Platform as a Service, mobile base station virtualization, and CDN virtualization. We believe that these tracks are vital and should be considered by fellow researchers working on NFV. In parallel to in-depth study of these use cases, more research should be dedicated to VM allocation, specifically to optimize some objectives which were neglected in comparison to others (e.g., overhead, ROI, and aggregate traffic).

Another point which is rarely discussed is the contextualization of service management. Logically, since virtualization enables cloud providers to manipulate directly the virtual equipment (i.e. quasi-inexistence of intermediary steps between the resources and targeted applications), the virtual equipment becomes somehow service agnostic, meaning that it

knows almost nothing about its contribution to the applications as a whole, which results in losing the management context capability [144].

Finally, although we can find that recently some researchers are working on the study of users' mobility and their service usage behavior, along with the corresponding placement of virtual appliances (i.e. VMs) required for VNFs [161], [165], [170], more research work must be carried out in order to allow a positioning of network functions in closer proximity to service generation, and consequently provide better QoE which would benefit both ISPs and the end users. Also, only a few VNFP solutions, such as in [123], [158], [166], are applied in multiple federated clouds, while most of the surveyed works treat the problem of VNFP within only a single cloud environment (e.g., in [42], [85], [86], [90].

XIV. Conclusion

Various aspects should be taken into consideration when seeking for an effective placement of virtualized network functions. These aspects include energy consumption, cost, performance degradation, SLA violations, and QoS. This paper investigates, in an extensive and detailed way, the existing VNF placement strategies and algorithms, organized in different categories, i.e. Network Functions chain placement, VNF forwarding graphs, VNF replications, ranging from generic VM-based NFV frameworks to VNF placement strategies for specific VNF types. This survey is meant to be a reference when investigating VNF and VM placement strategies. Relevant protocols, heuristics, algorithms, and architectures were surveyed with the main motivation to propose, as future work, efficient strategies to carry out efficient network slicing, in order to satisfy the end-users and verticals, and respect the several constraints in place.

Acknowledgment

This work was partially supported by the European Union's Horizon 2020 research and innovation programme under the 5G!Pagoda project with grant agreement No. 723172, and the Academy of Finland's Flagship programme 6Genesis (grant no. 318927).

References

[1] "Network Functions Virtualisation;An Introduction, Benefits, Enablers, Challenges & Call for Action," at the SDN and OpenFlow World Congress, Darmstadt-Germany, Oct. 2012.

[2] J. Gil Herrera and J. F. Botero, "Resource Allocation in NFV: A Comprehensive Survey," in IEEE Transactions on Network and Service Management, vol. 13, no. 3, pp. 518-532, Sep. 2016.

[3] ETSI GS NFV 001 : "Network Functions Virtualisation: Use Cases", ETSI Ind. Specification Group (ISG), Valbonne, France, Oct. 2013.

[4] ETSI GS NFV-SWA 001 : "Network Functions Virtualisation (NFV); Virtual Network Functions Architecture", ETSI Ind. Specification Group (ISG), Valbonne, France, Dec. 2014.

[5] ETSI GS NFV 002 : "Network Functions Virtualisation (NFV); Architectural Framework", ETSI Ind. Specification Group (ISG), Valbonne, France, Oct. 2013.

[6] ETSI GS NFV-IFA 002 : "Network Functions Virtualisation; Acceleration technologies; VNF Interfaces Specification", ETSI Ind. Specification Group (ISG), Valbonne, France, Mar. 2016.

[7] [Online; accessed 07-September-2018] https://www.openstack.org/videos/video/designing-for-nfv-lessons-learned-from-deploying-at-verizon

[8] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebar, I. Pratt and A. Warfield, "Xen and the Art of Virtualization, in Proceedings of the ACM Symposium on Operating Systems Principles (SOSP), vol.37, no. 5, pp. 164-177, Oct. 2003.

[9] B. Shen, S. Lee and S. Basu, "Caching strategies in transcoding-enabled proxy systems for streaming media distribution networks," in IEEE Transactions on Multimedia, vol. 6, no. 2, pp. 375-386, Apr. 2004.

[10] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt and A. Warfield, "Live Migration of Virtual Machines," in proc. of the 2nd Symposium on Networked Systems Design and Implementation (NSDI 05), Berkeley, CA, USA, pp. 273-286, May 2005.

[11] M. Nelson, B. Lim, and G. Hutchins, "Fast Transparent Migration for Virtual Machines Solutions Brief," in Proc. of USENIX 2005 General Track, Berkeley, CA, USA, pp. 25-25, May. 2005.

[12] C. Hyser, B. Mckee, R. Gardner, and B. J. Watson, "Autonomic virtual machine placement in the data center," in Hewlett Packard Laboratories, Tech. Rep. HPL-2007-189, pp. 1-10, Feb. 2008.

[13] Z. Wang, M. Marwah, C. Bash, P. Ranganathan, and X. Zhu, "Delivering Energy Proportionality with Non Energy-Proportional Systems – Optimizing the Ensemble," in Proc. of the Workshop on Power Aware Computing and Systems (HotPower '08), Dec. 2008.

[14] A. Singh, M. Korupolu, and D. Mohapatra, "Server storage virtualization: Integration and load balancing in data centers," in International Conference for High Performance Computing, Networking, Storage and Analysis, pp. 112, 2008.

[15] S. Chaisiri, B. S. Lee, and D. Niyato, "Optimal virtual machine placement across multiple cloud providers," in IEEE Asia-Pacific Services Computing Conference (APSCC), Singapore, pp. 103-110, Dec. 2009.

[16] H. N. Van, F. Dang Tran, and J. M. Menaud, "Autonomic virtual resource management for service hosting platforms," in Proc. of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing, IEEE Computer Society, Vancouver, BC, pp. 1-8, May. 2009.

[17] C. C. Yang, K. T. Chen, C. Chen and J. Y. Chen, "Market-Based Load Balancing for Distributed Heterogeneous Multi-Resource Servers," in 2009 15th International Conference on Parallel and Distributed Systems, Shenzhen, Dec. 2009, pp. 158-165.

[18] H. N. Van, F. D. Tran and J. Menaud, "SLA-Aware Virtual Resource Management for Cloud Infrastructures," 2009 Ninth IEEE International Conference on Computer and Information Technology, Xiamen, 2009, pp. 357-362.

[19] J. Xu and J. A. B. Fortes, "Multi-Objective Virtual Machine Placement in Virtualized Data Center Environments," Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom), Hangzhou, pp. 179-188, 2010.

[20] U.Bellur, C. S. Rao and S.D. M. Kumar, "Optimal Placement Algorithms for Virtual Machines," in CoRR, vol. abs/1011.5064, Nov. 2010.

[21] F. Machida, M. Kawato, and Y. Maeno, "Redundant virtual machine placement for fault-tolerant consolidated server clusters," in Network Operations and Management Symposium (NOMS), pp. 32-39, 2010.

[22] J. T. Piao and J. Yan, "A network-aware virtual machine placement and migration approach in cloud computing," in Grid and Cooperative Computing (GCC), 2010 9th International Conference on, pp. 87-92, 2010.

[23] X. Meng, V. Pappas, and L. Zhang, "Improving the scalability of data center networks with traffic-aware virtual machine placement," in proc. of INFOCOM, pp. 1-9, 2010.

[24] W. Emeneker, and A. Apon, "Cache effects of virtual machine placement on multi-core processors," in IEEE 10th International Conference on Computer and Information Technology (CIT), pp. 2261-2266, 2010.

[25] T. Cordeiro, D. Damalio, N. Pereira, P. Endo, A. Palhares, G. Goncalves, D. Sadok, J. Kelner, B. Melander, V. Souza, and J.-E. Mangs, "Open source cloud computing platforms," in 9th IEEE International Conference on Grid and Cooperative Computing, pp. 366-371, 2010.

[26] G. Y.Jung, K. R. Joshi, M. A. Hiltunen, R. D. Schlichting, and C.Pu, "Performance and availability aware regeneration for cloud based multitier applications," in Proc. of 2010 IEEE/IFIP International Conference on Dependable Systems and Networks, Chicago, IL, pp. 497-506, 2010.

[27] V. Cardellini, E. Casalicchio, F. Lo Presti and L. Silvestri, "SLA-aware Resource Management for Application Service Providers in the Cloud," in 2011 First International Symposium on Network Cloud Computing and Applications, Toulouse, pp. 20-27, 2011.

[28] D. Jayasinghe, C. Pu, T. Eilam, M. Steinder, I. Whally and E. Snible, "Improving Performance and Availability of Services Hosted on IaaS Clouds with Structural Constraint-Aware Virtual Machine Placement," 2011 IEEE International Conference on Services Computing, Washington, DC, 2011, pp. 72-79.

[29] D.Breitgand and A. Epstein, "SLA-aware placement of multivirtual machine elastic services in compute clouds," in 2011 IFIP/IEEE International Symposium on Network Management (IM 2011), pp. 161-168, 2011.

[30] K. Le, R. Bianchini, J. Zhang, Y. Jaluria, J. Meng and T. D. Nguyen, "Reducing electricity cost through virtual machine placement in high performance computing clouds," in Proc. of 2011 International Conference for High Performance computing, 2011, pp. 1-12.

[31] W. Shi and B. Hong, "Towards profitable virtual machine placement in the data center," in 2011 Fourth IEEE International Conference on Utility and Cloud Computing (UCC), pp. 138-145, 2011.

[32] C. C. T. Mark, D. Niyato, and T. Chen-Khong, "Evolutionary optimal virtual machine placement and demand forecaster for cloud computing," in IEEE International Conference on Advanced Information Networking and Applications (AINA), pp. 348-355, 2011.

[33] E. Bin, O. Biran, O. Boni, E. Hadad, E. K. Kolodner, Y. Moatti and D. H. Lorenz, "Guaranteeing high availability goals for virtual machine placement," in 31st International Conference on Distributed Computing Systems (ICDCS), pp. 700-709, 2011.

[34] C. C. Lin, P. Liu, and J. J. Wu, "Energy-efficient virtual machine provison algorithms for cloud systems," in 4th IEEE International Conference on Utility and Cloud Computing, pp. 81-88, 2011.

[35] J. Ousterhout, P. Agrawal, D. Erickson, C. Kozyrakis, J. Leverich, D. Mazieres, S. Mitra, A. Narayanan, D. Ongaro, G. Parulkar, M. Rosenblum, S. M. Rumble, E. Stratmann, and R. Stutsman, "The case for RAMCloud," in Commun. ACM, vol. 54, pp. 121-130, 2011.

[36] D. S. Dias and L. H. M. Costa, "Online traffic-aware virtual machine placement in data center networks," in Global Information Infrastructure and Networking Symposium (GIIS), pp. 1-8, 2012.

[37] B. C. Ribas, R. M. Suguimoto, R. A. Montano, F. Silva, L. de Bona, and M. Castilho, "On modelling virtual machine consolidation to pseudo boolean constraints," in Advances in Artificial Intelligence IBERAMIA 2012, Lecture Notes of Computer Science, vol. 7637, pp. 361-370, 2012.

[38] G. Wu, M. Tang, Y-C. Tian and Li W, "Energy-efficient virtual machine placement in data centers by genetic algorithm," in Proceeding of international conference on neural information processing, pp. 315-323, 2012.

[39] T. Knauth and C. Fetzer, "Energy-aware scheduling for infrastructure clouds," in 4th IEEE International Conference on Cloud Computing Technology and Science Proceedings, Taipei, pp. 58-65, 2012.

[40] M. Alicherry and T. V. Lakshman, "Network aware resource allocation in distributed clouds," 2012 Proceedings IEEE INFOCOM, Orlando, FL, pp. 963-971, 2012.

[41] C. Dupont, T. Schulze, G. Giuliani, A. Somov, and F. Hermenier, "An energy aware framework for virtual machine placement in cloud federated data centers," in Future Energy Systems: Where Energy, Computing and Communication Meet (e-Energy), 2012 Third International Conference on, pp. 1-10, 2012.

[42] G. Kim, H. Park, J. Yu, and W. Lee, "Virtual machines placement for network isolation in clouds," in Proc. of the 2012 ACM Research in Applied Computation Symposium, pp. 243-248, Oct. 2012.

[43] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers," in Proc. IEEE INFOCOM, pp. 1107-1115, 2012.

[44] Z. B. Zheng, T. C.Zhou, M. R. Lyu, and I. King, "Component ranking for fault-tolerant cloud applications," in IEEE Transactions on Services Computing, vol. 5, pp. 540-550, 2012.

[45] T. Taleb and A. Ksentini,"Follow me cloud: interworking federated clouds and distributed mobile networks," in IEEE Network, vol. 27, no. 5, pp. 12-19, 2013.

[46] X. Li, Z. Qian, S. Lu, and J. Wu, "Energy efficient virtual machine placement algorithm with balanced and improved resource utilization in a data center," in Mathematical and Computer Modelling, vol. 58, no. 5, pp. 1222-1235, 2013.

[47] B. C. Ribas, R. M. Suguimoto, R. A. Montano, F. Silva, and M. Castilho, "PBFVMC: A new pseudo-boolean formulation to virtual-machine consolidation," in Brazilian Conference on Intelligent Systems (BRACIS), pp. 201-206, 2013.

[48] A. Dalvandi, M. Gurusamy, and K. C. Chua, "Time-aware vm-placement and routing with bandwidth guarantees in green cloud data centers," in IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom), vol. 1, pp. 212-217, 2013.

[49] K. Li, J. Wu, and A. Blaisse, "Elasticity-aware virtual machine placement for cloud datacenters," in IEEE 2nd International Conference on Cloud Networking (CloudNet), pp. 99-107, 2013.

[50] A. C. Adamuthe, R. M. Pandharpatte, and G. T. Thampi, "Multiobjective virtual machine placement in cloud environment," in IEEE International Conference on Cloud and Ubiquitous Computing and Emerging Technologies (CUBE), pp. 8-13, 2013.

[51] Y. Gao, H. Guan, Z. Qi, Y. Hou, and L. Liu, "A multi-objective ant colony system algorithm for virtual machine placement in cloud computing," in Journal of Computer and System Sciences, vol. 79, no. 8, pp. 1230-1242, 2013.

[52] M. Alicherry and T. V. Lakshman, "Optimizing data access latencies in cloud systems by intelligent virtual machine placement," in Proc. IEEE INFOCOM, Turin, pp. 647-655, 2013.

[53] K. Kirkpatrick, "Software-defined networking," in Communications of the ACM, vol. 56, no. 9, pp. 16-19, 2013.

[54] [Online; accessed 07-September-2018] N. Feamster, "Software defined networking," in coursera: https://class. coursera.org/sdn-001. (2013).

[55] H. Kim, and N. Feamster, "Improving network management with software defined networking," in Communications Magazine, IEEE, vol. 51, no. 2, pp. 114-119, 2013.

[56] P. Farrow and M. Reed, "Optimising the geographical location of transcoding resources," in 2013 5th IEEE International Conference on Broadband Network & Multimedia Technology, Guilin, pp. 58-62, 2013.

[57] Z. Zhu, S. Li, and X. Chen, "Design QoS-aware multi-path provisioning strategies for efficient cloud-assisted SVC video streaming to heterogeneous clients," in IEEE Transactions on Multimedia, vol. 15, no. 4, pp. 758-768, 2013.

[58] A. Basta, W. Kellerer, M. Hoffmann, K. Hoffmann, and E. D. Schmidt, "A Virtual SDN-enabled LTE EPC Architecture: a case study for S-/P-Gateways functions," in IEEE SDN for Future Networks and Services (SDN4FNS), pp. 1-7, 2013.

[59] T. Taleb and A. Ksentini, "Gateway relocation avoidance-aware network function placement in carrier cloud," in Proceedings of the 16th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems, pp.341-346, 2013.

[60] A. K. Das, T. Adhikary, M. A. Razzaque and C. S. Hong, "An intelligent approach for virtual machine and QoS provisioning in cloud computing," in The International Conference on Information Networking 2013 (ICOIN), Bangkok, pp. 462-467, 2013.

[61] F. Z. Yousaf, J. Lessmann, P. Loureiro and S. Schmid, "SoftEPC Dynamic instantiation of mobile core network entities for efficient resource utilization," in 2013 IEEE International Conference on Communications (ICC), Budapest, pp. 3602-3606, 2013.

[62] Z. Cao and S. Dong, "An energy-aware heuristic framework for virtual machine consolidation in cloud computing," in The Journal of Supercomputing, pp. 1-23, 2014.

[63] M. Tang and S. Pan, "A hybrid genetic algorithm for the energy-efficient virtual machine placement problem in data centers," in Neural Processing Letters, pp. 1-11, 2014.

[64] S. H. Wang, P. P. W. Huang, C. H. P. Wen, and L. C. Wang, "Eqvmp: Energy-efficient and qos-aware virtual machine placement for software defined datacenter networks," in International Conference on Information Networking (ICOIN), pp. 220-225, 2014.

[65] X. Zhang, Q. Yue, and Z. He, "Dynamic energy-efficient virtual machine placement optimization for virtualized clouds," in Proceedings of the 2013 International Conference on Electrical and Information Technologies for Rail Transportation (EITRT2013), vol. II. Springer, pp. 439-448, 2014.

[66] N. T. Hieu and M. Di Francesco and A. Y. Jääski, "A virtual machine placement algorithm for balanced resource utilization in cloud data centers," in IEEE 7th International Conference on Cloud Computing (CLOUD), pp. 474-481, jun 2014.

[67] J. Kuo, H. Yang, M. Tsai, "Optimal approximation algorithm of virtual machine placement for data latency minimization in cloud systems," in Proceedings IEEE INFOCOM2014, pp. 1303-1311, 2014.

[68] F. Song, D. Huang, H. Zhou, H. Zhang, and I. You, "An optimization based scheme for efficient virtual machine placement," in International Journal of Parallel Programming, vol. 42, no. 5, pp. 853-872, 2014.

[69] S. Clayman, E. Maini, A. Galis, A. Manzalini and N. Mazzocca, "The dynamic placement of virtual network functions," in Network Operations and Management Symposium (NOMS), pp. 1-9, 2014.

[70] B. A.Nunes, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," in Communications Surveys & Tutorials, IEEE, vol. 16, no. 3, pp. 1617-1634, 2014.

[71] M. Bagaa, T. Taleb, and A. Ksentini, "Service-Aware Network Function Placement for Efficient Traffic Handling in Carrier Cloud," in Proc. IEEE WCNC'14, Istanbul, Turkey, Apr. 2014.

[72] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," in IEEE Communications Magazine, vol. 52, no. 8, pp. 82-89, 2014.

[73] H. Ahlehagh and S. Dey, "Video-aware scheduling and caching in the radio access network," in IEEE/ACM Transactions on Networking, vol. 22, no. 5, pp. 1444-1462, 2014.

[74] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. Leung, "Cache in the air: exploiting content caching and delivery techniques for 5G systems," in IEEE Communications Magazine, vol. 52, no. 2, pp. 131-139, 2014.

[75] H. Moens and F. D. Turck, "VNF-P: A model for efficient placement of virtualized network functions," in 10th International Conference on Network and Service Management (CNSM) and Workshop, pp. 418-423, 2014.

[76] A. Basta, W. Kellerer, M. Hoffmann, H. J. Morper, and K. Hoffmann, "Applying NFV and SDN to LTE mobile core gateways, the functions placement problem," in AllThingsCellular 14, pp. 33-38, 2014.

[77] T. Taleb, "Towards Carrier Cloud: Potential, Challenges, & Solutions," in IEEE Wireless Communications Magazine, vol. 21, no. 3, pp. 80-91, Jun. 2014.

[78] T. Taleb, M. Corici, C. Parada, A. Jamakovic, S. Ruffino, G. Karagiannis, and T. Magedanz, "EASE: EPC as a Service to Ease Mobile Core Network," in IEEE Network Magazine, Vol. 29, No. 2, pp.78-88, Mar. 2015.

[79] S. Herker, X. An, W. Kiess, S. Beker and A. Kirstaedter, "Data-Center Architecture Impacts on Virtualized Network Functions Service Chain Embedding with High Availability Requirements," 2015 IEEE Globecom Workshops (GC Wkshps), San Diego, CA, pp. 1-7, 2015.

[80] X. Fu and C. Zhou, "Virtual Machine Selection and Placement for dynamic consolidation in cloud computing environment," in Frontiers of Computer Science, Vol. 9, Issue 2, pp. 322-330, 2015.

[81] A. R. Ilkechi, I. Korpoeglu, O. Ulusoy, "Network-aware virtual machine placement in cloud data centers with multiple traffic-intensive components," in Computer Engineering Department, Bilkent University, Ankara Turkey 2015, vol. 91, pp. 508-527, 2015.

[82] T. Fukunaga, S. Hirahara and H. Yoshikawa, "Virtual Machine Placement for minimizing connection cost in Data Center Networks," in National Institute of Informatics, Japan, 2015.

[83] R. S. Moorthy,"A Strategy for Optimal Placement of Virtual Machines in IAAS Clouds," in International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), vol 4, Issue 4, 2015.

[84] D. Kreutz, F. M. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolky and S. Uhlig, "Software-defined networking: A comprehensive survey," in Proceedings of the IEEE, 103(1), pp. 14-76, 2015.

[85] F. Z. Yousaf, P. Loreiro, F. Zdarsky, T. Taleb, and M. Leibsch, "Cost Analysis of initial deployment strategies of a Virtual Network Infrastructure in a Datacenter," in IEEE Communications Magazine, Vol. 53, No. 12, pp. 60-66, Dec. 2015.

[86] T. Taleb, M. Bagaa, and A. Ksentini, "User Mobility-Aware Virtual Network Function Placement for Virtual 5G Network Infrastructure," in Proc. IEEE ICC 2015, London, UK, Jun. 2015.

[87] P. Farrow, M. Reed, M. Glowiak and J.Mambretti, "Transcoder Migration For Real Time Video Streaming Systems," in CoRR, vol.abs/1509.08091, 2015.

[88] M. Hajimirsadeghi, N. B. Mandayam, and A. Reznik, "Joint caching and pricing strategies for information centric networks," in Proc. IEEE Global Communications Conference (GLOBECOM), pp. 1-6, 2015.

[89] B. Addis, D. Belabed and M. Bouet, S. Secci, "Virtual Network Functions Placement and Routing Optimization," in CloudNet 2015, pp.171-177, 2015.

[90] S. Oechsner and A. Ripke, "Flexible support of VNF placement functions in OpenStack," Proceedings of the 2015 1st IEEE Conference on Network Softwarization (NetSoft), London, 2015, pp. 1-6.

[91] A. Baumgartner, V. S. Reddy, and T. Bauschert, "Mobile core network virtualization: A model for combined virtual core network function placement and topology optimization," in 1st IEEE Conference on Network Softwarization (NETSOFT). IEEE, pp. 1-9, 2015.

[92] R. Riggio, A. Bradai, T. Rasheed, J. Schulz-Zander, S. Kuklinski, and T. Ahmed, "Virtual Network Functions Orchestration in Wireless Networks," in 11th International Conference on Network and Service Management (CNSM), pp. 108-116, 2015.

[93] [Online; accessed 07-September-2018] Service chains with vSRX. http://www.juniper.net/techpubs/enUS/vsrx15.1x49/topics/concept/security-vsrx-contrail-service-chains.html

[94] X. Chen and J. Jiang, "A method of virtual machine placement for fault-tolerant cloud applications," in Intelligent Automation & Soft Computing, vol. 22, no. 4, pp. 587-597.

[95] Q. Zhenga, R. Lia, X. Lic, N. Shahd, J. Zhanga, F. Tiana, Kuo-M. Chaod and J. Lia, "Virtual machine consolidated placement based on multi-objective biogeography-based optimization," in Future Generation Computer Systems, vol. 54, pp. 95-122, 2016.

[96] B. Ahmad, T. Taleb, A. Vajda, and M. Bagaa, "Dynamic Cloud Resource Scheduling in Virtualized 5G Mobile Systems," in Proc. IEEE Globecom 2016, Washington, USA, Dec. 2016.

[97] B. Ahmad, A. Vajda, and T. Taleb, "Impact of Network Function Virtualization: A Study based on Real-Life Mobile Network Data," in Proc. IEEE IWCMC 2016, Paphos, Cyprus, Sep. 2016.

[98] J. Ortigoza, F. L. Pires and B. Baran, "Dynamic Environments for Virtual Machine Placement considering Elasticity and Overbooking," in CoRR, vol. abs/1601.01881, 2016.

[99] H. Routaib, E. Sabir, L. Badidi and M. Elkoutbi, "Latency Delay Evaluation for Cloudlet-based Architectures in Mobile Cloud Computing Environments," in Book entitled "Cloud and Fog Computing in 5G Mobile Networks: Emerging Advances and Applications", Mar. 2017.

[100] A. Zhou, S. Wang, B. Cheng, Z. Zheng, F. Yang, R. Chang, M. Lyu and R. Buyya, "Cloud Service Reliability Enhancement via Virtual Machine Placement Optimization," in IEEE Transactions on Services Computing, vol. 10, no. 6, pp. 902-913, Dec. 2017.

[101] A. Ksentini, M. Bagaa, T. Taleb, and I. Balasingham, "On using bargaining game for Optimal Placement of SDN controllers," in IEEE ICC'16, Kuala Lumpur, Malaysia, May. 2016.

[102] M. Hajimirsadeghi, N. B. Mandayam and A. Reznik, "Joint Caching and Pricing Strategies for Popular Content in Information Centric Networks," in IEEE Journal on Selected Areas in Communications, vol. 35, no. 3, pp. 654-667, Mar. 2017.

[103] A. Gharaibeh, A. Khreishah, B. Ji, and M. Ayyash, "A provably efficient online collaborative caching algorithm for multicell-coordinated systems," in IEEE Transactions on Mobile Computing, vol. 15, no. 8, pp. 1863-1876, 2016.

[104] T. X. Tran and D. Pompili, "Octopus: A Cooperative Hierarchical Caching Strategy for Cloud Radio Access Networks," in Proc. IEEE Int. Conf. on Mobile Ad hoc and Sensor Systems (MASS), Oct. 2016.

[105] T. X. Tran, A. Hajisami, and D. Pompili, "Cooperative Hierarchical Caching in 5G Cloud Radio Access Networks (C-RANs)," in CoRR, vol. abs/1602.02178, 2016.

[106] S. Mosleh, L. Liu, H. Hou, and Y. Yi, "Coordinated Data Assignment: A Novel Scheme for Big Data Over Cached Cloud-RAN," in Proc. IEEE Global Communications Conference (GLOBECOM), Dec. 2016.

[107] K. Poularakis, G. Iosifidis, A. Argyriou, I. Koutsopoulos, and L. Tassiulas, "Caching and operator cooperation policies for layered video content delivery," in Proc. IEEE INFOCOM, pp. 874-882, 2016.

[108] R. Yu, S. Qin, M. Bennis, X. Chen, G. Feng, Z. Han, and G. Xue, "Enhancing software-defined RAN with collaborative caching and scalable video coding," in Proc. IEEE ICC, pp. 1-6, 2016.

[109] T. Taleb and K. Hashimoto, "MS2: A Novel Multi-Source Mobile-Streaming Architecture," in IEEE Trans. on Broadcasting, Vol. 57, No. 3, pp. 662-673, Sep. 2011.

[110] S. Dutta, T. Taleb, P. A. Frangoudis, and A. Ksentini, "On-the-fly QoE-Aware Transcoding in the Mobile Edge," in Proc. IEEE Globecom 2016, Washington, USA, Dec. 2016.

[111] S. Dutta, T. Taleb, and A. Ksentini, "QoE-aware Elasticity Support in Cloud-Native 5G Systems," in IEEE ICC16, Kuala Lumpur, Malaysia, May. 2016.

[112] F. Ben Jemaa, G. Pujolle and M. Pariente, "QoS-Aware VNF Placement Optimization in Edge-Central Carrier Cloud Architecture," in IEEE Global Communications Conference (GLOBECOM), Washington, DC, pp. 1-7, 2016.

[113] F. Carpio, S. Dhahri and A. Jukan, "VNF Placement with Replication for Load Balancing in NFV Networks," in CoRR, vol. abs/1610.08266, 2016.

[114] H. A. Pedersen and S. Dey, "Enhancing mobile video capacity and quality using rate adaptation, RAN caching and processing," in IEEE/ACM Transactions on Networking, vol. 24, no. 2, pp. 9961010, 2016.

[115] T. X. Tran, P. Pandey, A.Hajisami and D. Pompili, "Collaborative Multi-bitrate Video Caching and Processing in Mobile-Edge Computing Networks," in CoRR, vol. abs/1612.01436, 2016.

[116] M. Mechtri, C. Ghribi, and D. Zeghlache. "VNF placement and chaining in distributed cloud". 9th IEEE International Conference on Cloud Computing," in IEEE Computer Society, Proceedings CLOUD 2016, pp.376-383, 2016.

[117] Q. Sun, P. Lu, W. Lu and Z. Zhu, "Forecast-Assisted NFV Service Chain Deployment Based on Affiliation-Aware vNF Placement," in IEEE Global Communications Conference (GLOBECOM), pp. 1-6, 2016.

[118] A. Laghrissi, S. Retal, and A. Idrissi, "Modeling and optimization of the network functions placement using constraint programming," in ACM International Conference Proceeding Series, Blagoevgrad, Bulgaria, no. 52, pp. 1-8, 2016.

[119] X. Li and C. Qian, "A survey of network function placement," in 2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, pp. 948-953, 2016.

[120] T. Taleb, K. Samdani, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Architecture & Orchestration, in IEEE Communications Surveys & Tutorials J., vol. 19, no. 3, pp. 1657-1681, May. 2017.

[121] T. Taleb, B. Mada, M. Corici, A. Nakao, and H. Flinck, "PERMIT: Network Slicing for Personalized 5G Mobile Telecommunications," in IEEE Communications Magazine, Vol. 55, No. 5, pp. 88-93, May. 2017.

[122] J. Cao, Y. Zhang, W. An, X. Chen, J. Sun, and Y. Han, "VNF-FG design and VNF placement for 5G mobile networks," in Science China Information Sciences, vol. 60, no. 4, Mar. 2017.

[123] A. Leivadeas, M. Falkner, I. Lambadaris, and G. Kesidis, "Optimal virtualized network function allocation for an SDN enabled cloud," in Computer Standards & Interfaces, vol. 54, no. 4, pp. 266-278, 2017.

[124] S. Khebbache, M. Hadji, and D. Zeghlache, "Virtualized network functions chaining and routing algorithms," in Computer Networks, vol. 114, pp. 95-110, 2017.

[125] 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," in Computer Communications, vol. 102, pp. 67-77, 2017.

[126] D. Bhamare, M. Samaka, A. Erbad, R. Jain, L. Gupta, and H. A. Chan, "Optimal virtual network function placement in multi-cloud service function chaining architecture," in Computer Communications, vol. 102, pp. 1-16, 2017.

[127] F. Carpio, W. Bziuk and A. Jukan, "Replication of Virtual Network Functions: Optimizing Link Utilization and Resource Costs," in CoRR, vol. abs/1702.07151, 2017.

[128] C. Pham, N. H. Tran, S. Ren, W. Saad and C. S. Hong, "Traffic-aware and Energy-efficient vNF Placement for Service Chaining: Joint Sampling and Matching Approach," in IEEE Transactions on Services Computing.

[129] D. Dietrich, C. Papagianni, P. Papadimitriou and J. S. Baras, "Network function placement on virtualized cellular cores," 2017 9th International Conference on Communication Systems and Networks (COMSNETS), Bangalore, pp. 259-266, 2017.

[130] S. Retal, M. Bagaa, T. Taleb, and H. Flinck, "Content Delivery Network Slicing: QoE and Cost Awareness," in Proc. IEEE ICC 2017, Paris, France, May. 2017.

[131] R. Mijumbi, J. Serrat, J. L. Gorricho, N. Bouten, F. De Turck and R. Boutaba, "Network Function Virtualization: State-of-the-Art and Research Challenges," in IEEE Communications Surveys & Tutorials, vol. 18, no. 1, pp. 236-262, 2016.

[132] A. Hmaity, M. Savi, F. Musumeci, M. Tornatore and A. Pattavina, "Virtual Network Function placement for resilient Service Chain provisioning," 2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM), Halmstad, pp. 245-252, 2016.

[133] J. Liu, W. Lu, F. Zhou, P. Lu and Z. Zhu, "On Dynamic Service Function Chain Deployment and Readjustment," in IEEE Transactions on Network and Service Management, vol. 14, no. 3, pp. 543-553, Sep. 2017.

[134] O. Brun, L. Wang and E. Gelenbe, "Big Data for Autonomic Intercontinental Overlays," in IEEE Journal on Selected Areas in Communications, vol. 34, No. 3, pp.575-583, 2016.

[135] Z. Usmani and S. Singh, "A Survey of Virtual Machine Placement Techniques in a Cloud Data Center," In Procedia Computer Science, Vol. 78, pp. 491-498, 2016.

[136] F. L. Pires and B. Barán, "Virtual Machine Placement Literature Review," in CoRR, vol.abs/1506.01509, 2015.

[137] Z. A. Mann. "Allocation of Virtual Machines in Cloud Data Centers - A Survey of Problem Models and Optimization Algorithms," in ACM Comput. Surv. 48, Article 11, pp. 1-34, Aug. 2015.

[138] D. Belabed, S. Secci, G. Pujolle and D. Medhi, "Striking a Balance Between Traffic Engineering and Energy Efficiency in Virtual Machine Placement," in IEEE Transactions on Network and Service Management, vol. 12, no. 2, pp. 202-216, Jun. 2015.

[139] S. Secci, P. Raad and P. Gallard, "Linking Virtual Machine Mobility to User Mobility," in IEEE Transactions on Network and Service Management, vol. 13, no. 4, pp. 927-940, Dec. 2016.

[140] M. Mangili, F. Martignon, and A. Capone, "Stochastic planning for content delivery: Unveiling the benefits of network functions virtualization," in Proc. IEEE ICNP, Raleigh, NC, USA, pp. 344-349, Oct. 2014.

[141] G. Peng, "CDN: Content Distribution Network," in CoRR, vol. cs.NI/0411069, 2004.

[142] T. Y. Kim and B. Lee, "Scalable CDN service PoC over distributed cloud management platform," in Proc. Int. Conf. Inf. Commun. Technol. Convergence (ICTC), Busan, South Korea, pp. 832-833, Oct. 2014.

[143] V. Verter, "Uncapacitated and Capacitated Facility Location Problems," in Eiselt H., Marianov V. (eds) Foundations of Location Analysis; International Series in Operations Research & Management Science, vol. 155, Springer, Boston, MA, pp. 25-37, 2011.

[144] M. J. Kim, H. G. Yoon, and H. K. Lee, "IMAV: An Intelligent Multi-Agent Model Based on Cloud Computing for Resource Virtualization," in Computers, Networks, Systems, and Industrial Engineering, pp. 99-111, 2011.

[145] M. Bouet, J. Leguay and V. Conan, "Cost-Based Placement of Virtualized Deep Packet Inspection Functions in SDN," in MILCOM 2013 - 2013 IEEE Military Communications Conference, San Diego, CA, pp. 992-997, 2013.

[146] M. Bouet, J. Leguay, T. Combe, and V. Conan, "Cost-based placement of vDPI functions in NFV infrastructures," in Int. J. Network Mgmt, vol. 25, pp. 490-506, 2015.

[147] ETSI GS NFV-SEC 002 V1.1.1 : "Network Functions Virtualisation (NFV); NFV Security; Cataloguing security features in management software", ETSI Ind. Specification Group (ISG), Valbonne, France, Aug. 2015.

[148] ETSI GS NFV-REL 002 V1.1.1 : "Network Functions Virtualisation (NFV); Reliability; Report on Scalable Architectures for Reliability Management", ETSI Ind. Specification Group (ISG), Valbonne, France, Sep. 2015.

[149] ETSI GS NFV-SEC 004 V1.1.1 : "Network Functions Virtualisation (NFV); NFV Security; Privacy and Regulation; Report on Lawful Interception Implications", ETSI Ind. Specification Group (ISG), Valbonne, France, Sep. 2015.

[150] M. Casazza, P. Fouilhoux, M. Bouet and S. Secci, "Securing Virtual Network Function Placement with High Availability Guarantees," in CoRR, vol. abs/1701.07993, 2017.

[151] A. J. Cahill and J. C. Sreenan, "An Efficient CDN Placement Algorithm for the Delivery of High-quality TV Content," in Proc. from Internet and Multimedia Systems and Applications, EuroIMSA, Grindelwald, 2005.

[152] H. Jiang, Z. Wang, A. K. Wong, J. Li and Z. Li, "A Replica Placement Algorithm for Hybrid CDN-P2P Architecture," in Proc. 15th International Conference on Parallel and Distributed Systems, Shenzhen, pp. 758-763, Dec. 2009.

[153] T. Wauters, J. Coppens, F. Turck, B. Dhoedt, and P. Demeester, "Replica placement in ring based content delivery networks," in Computer Communications, vol. 29, pp.3313-3326, 2005.

[154] "Hitachi Content Platform with Brocade vADC (solution profile)", Hitachi, Sep. 2016.

[155] [Online; accessed 07-September-2018] "Secure vADC Solutions", available: https://www.pulsesecure.net/vadc

[156] [Online; accessed 07-September-2018] "Application Delivery Controller Security: An Overview," available: https://kemptechnologies.com/quote-request/

[157] [Online; accessed 07-September-2018] "Whats the Difference Between Containers and Virtual Machines?," available: http://www.electronicdesign.com/dev-tools/what-s-difference-between-containers-and-virtual-machines

[158] M. Bagaa, T. Taleb, A. Laghrissi, and A. Ksentini, "Efficient Virtual Evolved Packet Core Deployment Across Multiple Cloud Domains," in Proc. IEEE WCNC 2018, Barcelona, Spain, Apr. 2018.

[159] W. John, F. Moradi, B. Pechenot and P. Skoldstrom, "Meeting the observability challenges for VNFs in 5G systems," in IFIP/IEEE Symposium on Integrated Network and Service Management (IM), Lisbon, pp. 1127-1130, 2017.

[160] DGS/NFV-066: "Network Functions Virtualisation (NFV); NFV Test; Report on CI/CD and Devops", ETSI Ind. Specification Group (ISG), Valbonne, France, Apr. 2017.

[161] A. Laghrissi, T. Taleb, M. Bagaa, and H. Flinck, "Towards edge slicing: VNF placement algorithms for a dynamic & realistic edge cloud environment," in 2017 IEEE Global Communications Conference, GLOBECOM 2017, Singapore, pp. 1-6, Dec. 2017.

[162] H. Huang, S. Guo, J. Wu and J. Li, "Service Chaining for Hybrid Network Function Clouds", IEEE Transactions on Cloud Computing, Jun. 2017.

[163] H. Huang, S. Guo, J. Wu and J. Li, "Green DataPath for TCAM-Based Software-Defined Networks," in IEEE Communications Magazine, vol. 54, no. 11, pp. 194-201, Nov. 2016.
[164] S. Fu, H. Wen, J. Wu and B. Wu, "Cross-Networks Energy Efficiency Tradeoff: From Wired Networks to Wireless Networks," in IEEE Access, vol. 5, pp. 15-26, Jun. 2016.
[165] A. Laghrissi, T. Taleb, and M. Bagaa, "Canonical domains for Optimal Network Slice Planning, in Proc. IEEE WCNC 2018, Barcelona, Spain, Apr. 2018.
[166] M. Bagaa, T. Taleb, A. Laghrissi, A. Ksentini and H. Flinck, "Coalitional Game for the Creation of Efficient Virtual Core Network Slices in 5G Mobile Systems," in IEEE Journal on Selected Areas in Communications, vol. 36, no. 3, pp. 469-484, Mar. 2018.
[167] J. Wu, "Green wireless communications: from concept to reality," in IEEE Wireless Communications, vol. 19, no. 4, Aug. 2012.
[168] Q. Luo, W. Fang, J. Wu, and Q. Chen, "Reliable broadband wireless communication for high speed train using baseband cloud," in EURASIP Journal on Wireless Communications and Networking, Vol. 12, No. 1, Sep. 2012.
[169] Y. Khettab, M. Bagaa, D. Dutra, T. Taleb, and N. Toumi, "Virtual Security as a Service for 5G Verticals," in Proc. IEEE WCNC 2018, Barcelona, Spain, Apr. 2018.
[170] A. Laghrissi, T. Taleb and M. Bagaa, "Conformal Mapping for Optimal Network Slice Planning Based on Canonical Domains," in IEEE Journal on Selected Areas in Communications, vol. 36, no. 3, pp. 519-528, Mar. 2018.
[171] I. Benkacem, T. Taleb, M. Bagaa and H. Flinck, "Optimal VNFs Placement in CDN Slicing Over Multi-Cloud Environment," in IEEE Journal on Selected Areas in Communications, vol. 36, no. 3, pp. 616-627, Mar. 2018.
[172] J. Wu, S. Guo, H. Huang, W. Liu and Y. Xiang, "Information and Communications Technologies for Sustainable Development Goals: State-of-the-Art, Needs and Perspectives." IEEE Communications Surveys & Tutorials, vol. 20, pp. 2389-2406, 2018.
[173] I. Benkacem, T. Taleb, M. Bagaa, and H. Flinck, "Performance Benchmark of Transcoding as a Virtual Network Function in CDN as a Service Slicing," in Proc. IEEE WCNC 2018, Barcelona, Spain, Apr. 2018.
[174] I. Afolabi, T. Taleb, K. Samdanis, A. Ksentini, and H. Flinck, "Network Slicing & Softwarization: A Survey on Principles, Enabling Technologies & Solutions," in IEEE Communications Surveys & Tutorials.
[175] J. Wu, S. Guo, J. Li and D. Zeng, "Big Data Meet Green Challenges: Big Data Toward Green Applications," in IEEE Systems Journal, vol. 10, no. 3, pp. 888-900, Sep. 2016.
[176] C. Ge, Z. Sun, N. Wang, K. Xu and J. Wu, "Energy Management in Cross-Domain Content Delivery Networks: A Theoretical Perspective," in IEEE Transactions on Network and Service Management, vol. 11, no. 3, pp. 264-277, Sep. 2014.

BIOGRAPHIES

Abdelquoddouss Laghrissi (abdelquoddouss.laghrissi@aalto.fi) received the bachelor's degree in mathematics and computer science and the master's degree in applied computer science with a dissertation on empathy in vehicular ad hoc networks from the School of Sciences, Mohamed V, Morocco. University, in 2012 and 2014, respectively. He is currently pursuing his Ph.D. degree with the School of Electrical Engineering, Aalto University, Finland. From 2014 to 2015, he was a Volunteer Member of the cloud computing working group within a Euro-Mediterranean project MOSAIC on Cooperation with Mediterranean Partners to build Opportunities around ICT and Societal and Industrial Challenges of H2020, and is currently involved in a European project 5G!Pagoda on network slicing and 5G in the context of H2020, during which he published articles and contributed to several deliverables. He is a member of the MOSA!C Lab. His research interests include mobile cloud computing, network function virtualization, and software defined networking.

Tarik Taleb (tarik.taleb@aalto.fi) is currently a professor at the School of Electrical Engineering, Department of Communications and Networking, Aalto University, 06220 Espoo, Finland. Before, he worked as a senior researcher and 3GPP standards expert at NEC Europe Ltd. Prior to his work at NEC, and until March 2009, he worked as assistant professor at the Graduate School of Information Sciences, Tohoku University, Japan, in a lab fully funded by KDDI. He received his B.E. degree in information engineering with distinction, and his M.Sc. and Ph.D. degrees in information sciences from Tohoku University in 2001, 2003, and 2005, respectively. His research interests lie in the field of architectural enhancements to mobile core networks (particularly 3GPPs), 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. He is a member of the IEEE Communications Society Standardization Program Development Board and serves as Steering Committee Chair of the IEEE Conference on Standards for Communications and Networking. He has received many awards for his many contributions in the area of mobile networking.