Resource Allocation in NFV: A Comprehensive Survey

Juliver Gil Herrera and Juan Felipe Botero

***Abstract*—Network functions virtualization (NFV) is a new network architecture framework where network functions that traditionally used dedicated hardware (middleboxes or network appliances) are now implemented in software that runs on top of general purpose hardware such as high volume servers. NFV emerges as an initiative from the industry (network operators, carriers, and manufacturers) in order to increase the deployment flexibility and integration of new network services with increased agility within operator's networks and to obtain significant reductions in operating expenditures and capital expenditures. NFV promotes virtualizing network functions such as transcoders, firewalls, and load balancers, among others, which were carried out by specialized hardware devices and migrating them to software-based appliances. One of the main challenges for the deployment of NFV is the resource allocation of demanded network services in NFV-based network infrastructures. This challenge has been called the NFV resource allocation (NFV-RA) problem. This paper presents a comprehensive state of the art of NFV-RA by introducing a novel classification of the main approaches that pose solutions to solve it. This paper also presents the research challenges that are still subject of future investigation in the NFV-RA realm.**

***Index Terms*—Network function virtualization, virtual network functions, resource allocation, NFV orchestration, VNF forwarding graph, scheduling, service chaining and placement.**

NOMENCLATURE

BSS	Business Support System
CAPEX	Capital Expenditures
EMS	Element Management System
ETSI	European Telecom Standards Institute
HVS	High Volume Server
ILP	Integer Linear Programming
NF	Network Function
NFV	Network Functions Virtualization
NFV-MANO	NFV Management and Orchestration
NFV-RA	NFV Resource Allocation
NFVI	Network Functions Virtualization Infrastructure
NS	Network Service
OPEX	Operating Expenditures
OSS	Operations Support System
SDN	Software Defined Networking
SLA	Service Level Agreement
SN	Substrate Network
TSP	Telecom Service Provider
VM	Virtual Machine
VNF	Virtual Network Function
VNF-FG	Virtual Network Function Forwarding Graph
VNF-FGE	VNF Forwarding Graph Embedding
VNFR	Virtual Network Functions Request
VNFs-CC	VNFs Chain composition
VNFs-SCH	VNFs Scheduling.

Manuscript received February 21, 2016; revised June 5, 2016; accepted August 1, 2016. Date of publication August 5, 2016; date of current version September 30, 2016. This research has been supported by COLCIENCIAS and the CODI project 2014-856 of the University of Antioquia. The associate editor coordinating the review of this paper and approving it for publication was C. Esteve Rothenberg.

The authors are with the Faculty of Engineering, University of Antioquia, Medellín 1226, Colombia (e-mail: juliver.gil@udea.edu.co; juanf.botero@udea.edu.co).

Digital Object Identifier 10.1109/TNSM.2016.2598420

I. INTRODUCTION

OFTEN, carrier networks are forced to substantially increase both CAPEX and OPEX when they deploy or update their physical network infrastructure, due to the need of specialized network hardware (known as middleboxes or hardware appliances), to offer new network services [1], [2]. Multimedia caches, QoS monitors, video transcoders, gateways and proxies are examples of middleboxes used by carrier and network operators to meet Service Level Agreements (SLAs). However, the use of middleboxes to provide new services suffers of several shortcomings: they *i*) are expensive, *ii*) require specialized managing personnel, *iii*) have high energy costs, *iv*) do not allow to add new functionality and *v*) have short lifecycles. Recent studies show that, in an enterprise network, the number of middleboxes is comparable with the number of routers and switches needed to maintain the operation of the network [3].

NFV emerges from the industry and promises to solve the aforementioned inconveniences, thus avoiding the constant proliferation of hardware appliances. It also facilitates and promotes innovation in the network by leveraging virtualization technology to offer a new way to design the networks [4], [5]. In November 2012, seven of the world's leading telecom network operators selected the European Telecom Standards Institute (ETSI) [6] to be the home of the industry specification group for NFV. Under the paradigm of NFV, traditional middleboxes are managed as single modules of software, programmed to play the role of a particular Virtual Network Function (VNF), this allows modularity and isolation of each function, so they can be managed independently. In addition, NFV facilitates installation and deployment of VNFs on general purpose servers (e.g., x86-based blades) [7], [8],

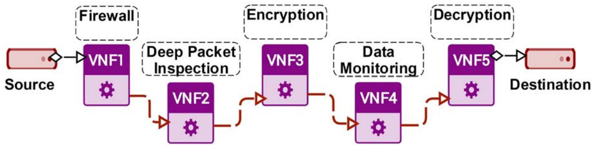

Fig. 1. Service Chain.

thus allowing dynamic migration of VNFs from one server to another, that is, to any place of the network [9].

In summary, NFV is the envisioned framework to solve most of the current network problems due to the wide use of specific hardware appliances. Also, it provides opportunities for network optimization and cost reduction. Moreover, it enables to configure hybrid scenarios where functions running on virtualized resources co-exist with those running on physical resources [10]. Such hybrid scenarios may be important in the transition towards NFV.

The traditional deployment of a Network Service (NS) requires the data traffic to pass through a certain fixed set of middleboxes in a particular order, which cause some processing according to the function they perform. The task of choosing the needed middleboxes and steer the traffic among them is commonly known as middleboxes orchestration [11]. Currently, this task is performed manually, and is set at the forwarding table entries of routers; the above is a cumbersome and error prone process [12]. Moreover, any placement of these physical middleboxes is destined to become ineffective over time; because it is very costly and impractical to keep changing the location of these hardware with changing network conditions.

In the NFV ecosystem, an Network Service (NS) is a set of chained VNFs as shown in Fig. 1. An NS is built and deployed in NFV by defining its: *i*) number of VNFs, *ii*) their respective order in the chain and *iii*) the allocation of the chain in the Network Functions Virtualization Infrastructure (NFVI), also called Substrate Network (SN).[1]

One of the main challenges to deploy NFV is to achieve fast, scalable and dynamic composition and allocation of NFs to execute an NS. However, since an NS requires a set of VNFs, achieving an efficient services' coordination and management in NFV raises two questions: 1) how to compose VNFs for a determined NS, and 2) how to efficiently allocate and schedule the VNFs of an NS onto a SN. The ETSI, through its NFV technologies group, is partnering with network operators and equipment vendors to promote NFV and is currently progressing with regard to the first question above.

This paper gives a context on the promising NFV technology and widely addresses the resource allocation challenge in NFV-based networks. Here, a classification of the NFV-RA problem by considering all its variants, namely; stages (VNFs Chain Composition, Embedding and Scheduling), optimization objectives, solution strategy and application domain.

The paper is organized as follows. Section II briefly describes the main features of NFV. Section III details the resource allocation problem in the context of NFV, and introduces its stages: *VNF-Chain Composition*, *Embedding* and *Scheduling*. An extensive classification of existing approaches to solve the NFV-RA is given in Section IV. Section V identifies the main emerging research directions within the NFV-RA field. Finally, Section VI concludes the paper.

[1]Here, the terms Substrate Network, Network Functions Virtualization Infrastructure and Physical Network are used interchangeably.

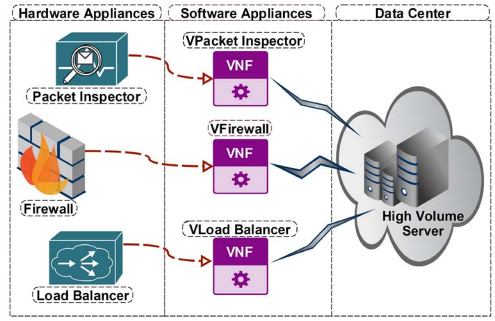

Fig. 2. NFV Environment.

II. NFV in a Nutshell

In this section, we provide a short background on NFV, including relevant aspects such as its architectural framework. In addition, we introduce the resource allocation problem in the NFV environment.

A. Purposes and Use Cases

Network functions virtualization is a recent initiative from the industry [10], [13], [14]. It emerges due to the vast and growing amount of proprietary and specific hardware currently deployed in network operators and the costs that it entails.

Offering a new network service often requires to install new proprietary hardware implying to find new physical space and to incur in new acquisition, installation, operation and energy costs. These costs added to the training of personnel able to design, integrate and operate the increasing number of heterogeneous hardware are making it difficult and unprofitable for the network operators to develop innovative network services. Additionally, these proprietary equipment quickly reach the end of their useful life, which leads to the cycle of *i)* acquisition, *ii)* integration and *iii)* deployment, constantly repeated with little economic benefit to network operators. This restricts inventiveness, in a world eager to receive new services focused on the Internet.

The goal of NFV is transforming the way network operators and network providers design, manage and deploy their network infrastructure thanks to the evolution of virtualization technologies. This transformation is being performed by means of the consolidation of different VNFs types in standard general purpose computers (servers, storage devices, etc.), which may be located in data centers, network nodes and close to end user premises (see Fig. 2) [15]. Even open source approaches are being developed in the field of NFV [16].

It is worth highlighting that the general concept of decoupling NFs from dedicated hardware does not necessarily require virtualization of resources. This means that the Telecom Service Provider (TSP) could still purchase or develop software (NFs) and run it on physical machines. The difference is that these NFs would have to be able to run on commodity servers [17]. However, the gains (such as flexibility, dynamic resource scaling, energy efficiency) anticipated from running these functions on virtualized resources are very strong selling points of NFV. Needless to mention, it is also possible to have hybrid scenarios as it was previously mentioned.

The majority of the network nodes and functions may be considered for virtualization but, in order to span the scope of technical challenges, ETSI has selected a set of relevant use cases [18]:

* *Network Functions Virtualization as a service*: NFV infrastructure, platform and even a single VNF instance can be provided as a service by a TSP, based on models similar to the cloud computing service models [19].

* *Virtualization of Mobile Core Network and IMS*: Mobile networks and the IP Multimedia Subsystem are populated with a large variety of proprietary hardware appliances, which costs and complexity can be reduced introducing NFV (specially for the coming 5G) [20]–[23].

* *Virtualization of Mobile Base Station*: Mobile operators can apply NFV in order to reduce costs as well as continuously develop and provide better service to their customer [24].

* *Virtualization of the Home Environment*: Installation of new equipment can be avoided in the home environment with the introduction of VNFs, reducing maintenance and improving service provision [25].

* *Virtualization of CDNs*: Content Delivery Networks use cache node to improve the quality of multimedia services, but it comes with lots of disadvantages (e.g., waste of dedicated resources) that could be mitigated by NFV [26], [27].

* *Fixed Access Network Functions Virtualization*: virtualization supports multiple tenancy in access network equipment, whereby more than one organizational entity can either be allocated, or given direct control of, a dedicated partition of a virtual access node [28], [29].

B. Architecture

VNFs can be deployed and reassigned to share different physical and virtual resources of the infrastructure, so as to guarantee scalability and performance requirements. This allows TSPs to rapidly deploy new and elastic services [9], [30]. In general, there are three main components in the NFV architecture: Services, NFVI and the NFV Management and Orchestration (NFV-MANO), as shown in Fig. 3.

These components are described as follows.

1) Services: A service is a set of VNFs, that can be implemented in one or multiple virtual machines. In some situations, VNFs can run in virtual machines installed in operating systems or on the hardware directly; they are managed by native hypervisors or virtual machine monitors.

A VNF is usually administered by an Element Management System (EMS), responsible of its creation, configuration, monitoring, performance and security. An EMS provides the essential information required by the Operations Support System (OSS) in a TSP's environment.

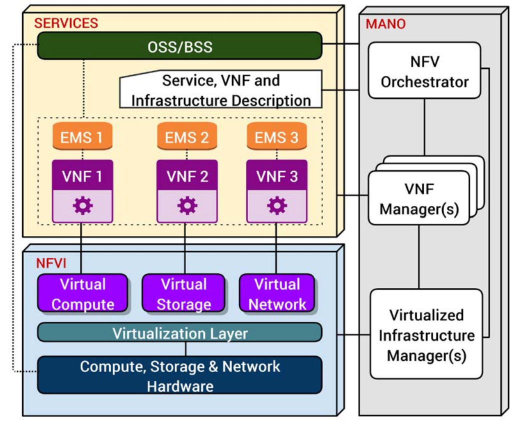

Fig. 3. ETSI-NFV Architecture.

The OSS is the general management system, that, along with the Business Support System (BSS), help providers to deploy and manage several end-to-end telecommunications services (e.g., ordering, billing, renewals, problem troubleshooting, etc.). NFV specifications focus on integration with existing OSS/BSS solutions [31].

2) NFVI: NFV infrastructure covers all hardware and software resources that comprise the NFV environment. NFVI includes network connectivity between locations, e.g., between data centers and the public or private hybrid clouds. Physical resources typically include computing, storage and network hardware providing processing, storage and connectivity for VNFs through the virtualization layer that sits just above the hardware and abstracts the physical resources (logically partitioned and assigned to VNFs).

There is no specific solution for the deployment of NFV; rather NFV architecture can take advantage of an existing virtualization layer, such as a hypervisor, with standard features that simply extracts the hardware resources and assigns them to the VNFs. When this support is not available, often, the virtualization layer is achieved through an operating system that adds software on top of a non-virtualized server or by implementing a VNF as an application [9].

3) NFV-MANO: NFV Management and Orchestration is composed of: the orchestrator, VNFs managers and Virtualized Infrastructure Managers. Such blocks provide the functionality required for the management tasks applied to the VNFs, e.g., provisioning and configuration.

NFV-MANO includes the orchestration and lifecycle management of physical or virtual resources that support the infrastructure virtualization, and the lifecycle management of VNFs. It also includes databases that are used to store the information and data models defining both deployment as well as lifecycle properties of functions, services, and resources.

NFV-MANO focuses on all virtualization-specific management tasks necessary in the NFV framework. In addition, the framework defines interfaces that can be used for communications between the different components of the

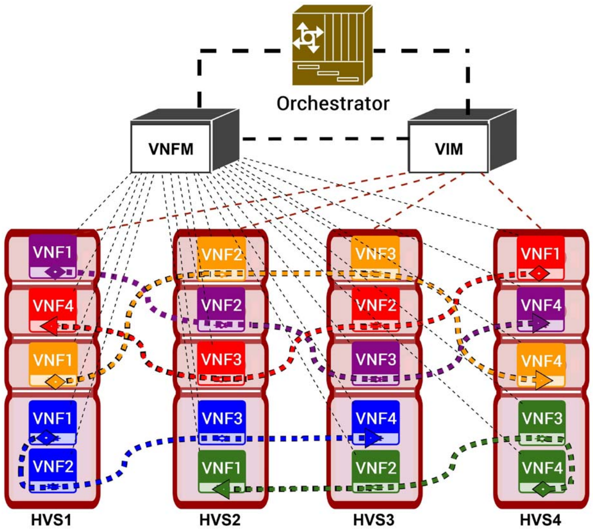

Fig. 4. NFV Management and Orchestration.

NFV-MANO, as well as coordination with traditional network management systems (i.e., OSS and BSS) to allow the operation of both VNFs and functions running on legacy equipment [10], [31].

Summarizing, if an NS using a firewall and a DPI is deployed, then NFV-MANO shall be responsible to say where these VNFs are located on the physical network. In turn, these VNFs are controlled by the EMS and the same MANO. Besides, the virtualization layer exposes the physical resources of chosen NFVI locations to the VNFs.

C. Relationship Between NFV and Software Defined Networks

NFV is closely related to other emerging technologies, such as Software Defined Networking (SDN) [32]. SDN is a networking technology that decouples the control plane from the underlying data plane and consolidates the control functions into a logically centralized controller.

NFV and SDN are mutually beneficial, highly complementary to each other, and share the same feature of promoting innovation, creativity, openness and competitiveness [33], [34]. For example, SDN can support NFV to enhance its performance, facilitate its operation and simplify the compatibility with legacy deployments. However, we emphasize that the virtualization and deployment of network functions do not necessarily rely on SDN technologies, and viceversa [4], [35].

D. Resource Allocation Problem in NFV Architecture

Resource allocation in NFV requires efficient algorithms to determine on which HVSs VNFs are placed, and be able to migrate functions from one server to another for such objectives as load balancing, reduction of CAPEX and OPEX, energy saving, recovery from failures, etc. [36].

In the NFV architecture framework the component that performs the resource allocation is the orchestrator. Fig. 4 illustrates a scenario where the orchestrator manages VNFs through the VNF manager and the virtualized infrastructure manager (see Fig. 3). The orchestrator evaluates all the conditions to perform the assignment of VNFs chains on the physical resources, leaning on the VNF managers and the virtualized infrastructure managers. The resource allocation in NFV is carried out in three stages: 1) VNFs Chain composition (VNFs-CC), also known in the literature as Service Function Chaining [37]–[40] 2) VNF Forwarding Graph Embedding (VNF-FGE)[2] and 3) VNFs Scheduling (VNFs-SCH). Next section deeply details the NFV-RA problem and its derived sub-stages.

If the reader wants to expand his knowledge on NFV, please refer to [10].

III. NFV-Resource Allocation

This section is divided into three parts. First, we take a look at the well-known Virtual Network Embedding (VNE) problem and explain how it relates to the NFV-RA problem. The second part gives a brief conceptual definition of VNFs and services. The third part deeply elaborates on the NFV-RA problem describing in detail each of its stages.

A. VNE Problem

Embedding virtual networks in a SN is the main resource allocation challenge in network virtualization and is usually referred to as the VNE problem [41], [42]. VNE deals with the allocation of virtual resources both in nodes (mapped to substrate nodes) and links (mapped to substrate paths) [43]. It is mainly concerned with the efficient mapping Virtual Network Requests onto a shared substrate network.

The VNE problem can be either offline or online. In offline problems, all the virtual network requests are known and scheduled in advance while for the online problem, such requests arrive dynamically and can stay in the network for an arbitrary duration [44]–[46]. VNE is known to be $\mathcal{NP}$-hard [47]; therefore, most of the work done in this area has focused on the design of heuristic or metaheuristic algorithms and the use of networks with minimal complexity when solving mixed Integer Linear Programming (ILP) models.

Embeddings can be optimized with regard to several parameters, such as: embedding cost, link bandwidth, QoS, economical profit, network survivability, energy efficiency [48], [49], security [50], [51], etc.

VNE and NFV-RA are in the same problem domain; in the end, the outcome of both problems is the efficient allocation of virtual requests on top of the physical network infrastructure. However, they present the following differences:

- VNE has static virtual network topologies where nodes are arranged in a fixed, predetermined order as input. In contrast, NFV-RA's input is a network service request composed of a set of VNFs with precedence constraints and resource demands that can be denoted by several Virtual Network Function Forwarding

[2]The terms embedding and mapping are used synonymously.

Graphs (VNF-FGs)[3]; the task of the first stage of the problem (VNFs-CC) is to efficiently build a suitable VNF-FG with regard to the operator's goals.

- Resource demands may change depending of the traffic load directed to them: e.g., computing resource demands of a transcoding VNF vary depending on how many multimedia data have to be transcoded. Also, bandwidth demands change depending on the ordering of VNF instances [52], whereas resource demands are mostly static in VNE.
- VNE shares several similarities with the second stage of NFV-RA: VNF-FGE. In fact, we think of VNF-FGE as a generalization of VNE because, while the latter considers only one type of physical (*networking*) device, a much wider number of different network functions coexist in NFV that can be mapped in different kind of hardware (*networking*, *computation* and *storage*) devices.

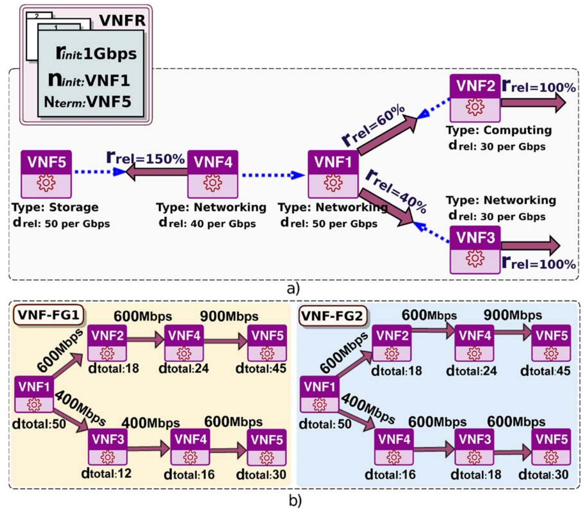

Fig. 5. VNFs Chain Composition.

B. VNFs and Services

A NF is a functional block within a physical network infrastructure that has well defined external interfaces and functional behaviour [53]. Examples of NFs are elements in a home network, e.g., Residential Gateway; and conventional network functions, e.g., DHCP servers, firewalls, etc.

In NFV, a NF is built as a software module that can be virtualized and deployed on a Virtual Network Infrastructure (VNI). A single VNF may be composed of multiple internal components, and hence it could be deployed over multiple Virtual Machines (VMs), in which case each VM hosts a single component of the VNF [9].

A NS is composed of one or more NFs. In the case of NFV, the NFs that make up the service are virtualized and deployed on virtual resources such as a VMs hosted on high volume server (HVS)s. However, in the perspective of the users, the services, whether based on functions running dedicated equipment or on VMs, should have the same performance.

C. Introduction to NFV-RA

A comprehensive description about the three stages that conform the resource allocation problem in NFV-based network's architectures is presented below.

1) VNFs - Chain Composition (VNFs-CC): NFV exploits the flexibility introduced by virtualization to dynamically compose chains of VNFs and strategically deploy them on a set of physical network nodes so as to achieve a predefined operator's objective or to meet an SLA, unlike the current static network function chain placement that depends on the physical location of the middleboxes in the SN [9].

The ETSI defines an NS as entities composed by an ordered number of VNFs [9]. That is, a packet must pass through a set of VNFs to be part of the offered network service. As VNFs are software, one of the main challenges that arises is: How to concatenate the different VNFs efficiently in order to compose an NS in the most adequate way, with respect to the TSP goals? This first challenge is the chaining process, that we call *chain composition*. TSPs will need to efficiently compose such chains to deploy customized and dynamic NFV-enabled network services [54], [55].

Fig. 5 illustrates VNFs-CC. It shows a Virtual Network Functions Request (VNFR)[4] and two possible chainings (VNF-FGs) of its VNF instances. For the i-th VNFR, VNFR^i, the initial data rate of the network flow -$r_{\text{init}}(\text{VNFR}^i)$- and its VNFs are given. Some VNFs may split the traffic flow; for instance, a load balancer VNF separating incoming data into two streams can specify that 60% of the incoming traffic is forwarded to VNF 2 and 40% to VNF 3 (see VNF 1, 2, and 3 in Fig. 5a). $l^i(\text{VNF})$ denotes the set of "outgoing" links of a VNF. For each link, relative traffic rate -r_{rel}- percentage with respect to the total outgoing VNF's traffic is defined (60% and 40% in the case of VNF 1).

Neither bandwidth demands of the links nor capacity demands of the nodes in the VNF-FG are static; they depend on the ordering of the VNFs. This ordering is flexible, but it is tied to the dependencies between VNFs; i.e., the network flow first has to pass through a set of VNFs before it arrives at a specific VNF (blue dotted line in Fig. 5a from VNF 4 to VNF 1, implying that VNF 4 depends on VNF 1 and must therefore be executed after VNF 1). Additionally, VNF link dependencies can be defined for VNFs that should selectively be placed on one of the sub-flows (see VNFs 2 and 3, both pointing to the VNF links of VNF 1 in Fig. 5a).

Based on the dependencies, several valid chaining options (VNF-FGs) of VNFRs can be derived. Fig. 5b shows two possible VNF-FGs for the VNFR in Fig. 5a). In VNF-FG 1, the 1 GBps (r_{init}) is divided in 600 MBps (60%) to VNF 2 and 400 MBps (40%) to VNF 2 as indicated in the VNFR. In VNF-FG 2, the composition of the VNFs is different as in the lower branch VNF 4 goes before VNF3, which is another perfectly feasible chaining solution (see Fig. 5b).

[3] As defined by ETSI, VNF-FG is a graph that describe how the flow should traverse the service through the network functions.

[4] Here, VNFR and Network Service Request are used synonymously.

Moreover, the amount of required processing capacities is specified for handling the network flow. E.g., a VNF performing video encoding should always be embedded on top of a computing node and demands 500 MHz of CPU processing capacities to encode 100 MBit/s. The amount of required capacities depends on the amount of data handled by that VNF instance. Fig. 5b shows that relative data rate d_{rel} of VNF 3 is 30 units per GBps, then total data rate in the VNF-FG is $d_{\text{total}} = 30 * 0.4 = 12$.

For each VNF, one or more subsequent VNF instances are created. Multiple VNF instances of the same VNF need to be created in scenarios where the network flow is split and traffic has to be processed by the same *type* of VNFs, even if traffic is not routed through the same VNF *instance* (in Fig. 5b, both chains require two instances of VNF 4).

In this regard, performance of NSs will be affected by both the different composing functions' behavior and the order in which functions are processed. Therefore, it is paramount to achieve an efficient service chain composition with regard to network operator's objectives.

Up to now, most NFV-RA proposals consider the VNF-FG as an input of the problem, i.e., the chain composition stage is neglected. Few approaches have been proposed to solve chain composition stage of NFV-RA: Mehraghdam *et al.* [56] formulate a context-free language for formalizing the chaining requests. In addition, they propose a greedy heuristic that tries to minimize the total data rate of the resulting chain by sorting the VNFs in ascending order according to their ratio of outgoing to incoming data rate and trying to chain first the VNF that reduces the data rate of the flows the most in each step. Recently, Beck and Botero [52] proposed a scalable recursive heuristic that, at each step, compose a VNF in the service chain and, at the same time, embeds it in the SN. The approach recursively maps the VNFs one by one and, when a VNF cannot be embed, performs backtracking by going back to the last successfully mapped VNF and looks for a different mapping alternative in order to rapidly find a feasible solution.

2) VNF - Forwarding Graph Embedding (VNF-FGE): As it has been previously stated, the chain of VNFs composing an end-to-end network service is called VNF-FG. This resulting graph in the first stage is given as the input of the embedding stage. VNF-FG is composed by the ordered set of VNFs that the NS runs in order to fulfill service's attributes (e.g., reliability, availability, security and performance) [53].

VNF-FGE is the second challenge identified in the NFV-RA, which seeks to find *where* to allocate the VNFs in the network infrastructure in a suitable way, considering a set of requested network services. Besides, resource optimization must be accomplished with regard to a specific objective (e.g., maximization of remaining network resources, minimization of SN's power consumption, optimization of a specific QoS metric, etc.). As it has been previously discussed, VNF-FGE can be seen as a generalization of the well-known VNE or VDCE problem [42], [57]–[59].

VNE is $\mathcal{NP}$-hard [47] and, as a VNF-FGE is a generalization of VNE, it is also $\mathcal{NP}$-hard. Generally speaking, the problem consists on the mapping of virtual resources to candidate substrate resources. Only if all virtual resources can be mapped, the entire network is then embedded and substrate resources are actually spent. The two stages of VNE also apply to VNF-FGE: virtual node and virtual link mapping. In addition, in VNF-FGE, each VNF is annotated with a type: computing, storage or networking and therefore, it has to be allocated into a physical node that meet the VNF's type.

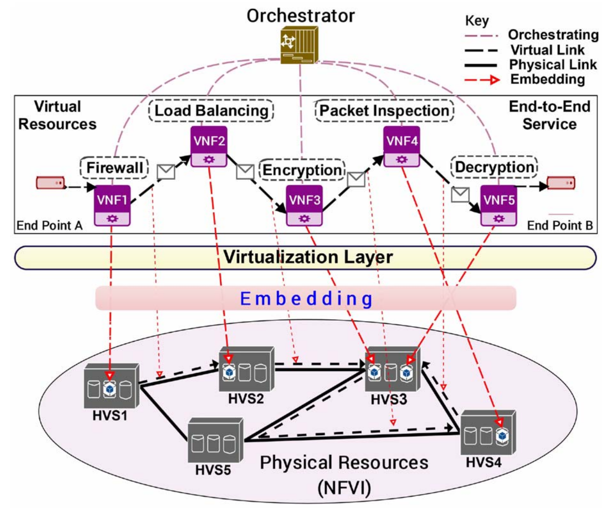

Fig. 6. VNFs forwarding-graph embedding.

Fig. 6 shows the deployment of an end-to-end network service $\boldsymbol{S} = \{Firewall \Rightarrow LoadBalancing \Rightarrow Encryption \Rightarrow PacketInspection \Rightarrow Decryption\}$ between two substrate nodes in a NFV-enabled SN infrastructure. It illustrates the embedding stage involving: Orchestrator, service, virtualization layer and NFVI. The orchestrator is responsible of the embedding stage; it is in charge of the management and orchestration of software resources and the virtualized hardware infrastructure to realize networking services. The service is composed by a set of VNFs that together provides a specific functionality. The virtualization layer abstracts the physical resources and anchors the VNFs to the virtualized infrastructure. It ensures that the VNF lifecycle is independent of the underlying hardware platforms by offering standardized interfaces [4]. This type of functionality is typically provided in the forms of VMs and their hypervisors which can be located in data centers, at network nodes, and in end-user facilities [60].

A HVS is considered a physical network node in a NFV-based network architecture, which uses a hypervisor to manage virtual machines, according to the availability of resources *(CPU, Disk, NIC and RAM)*. VMs running on top of HVSs can host one or more VNFs of the same type (computing, storage, networking).

An example to illustrate the embedding stage is shown in Fig. 6. First of all, it is important to clarify that the orchestrator runs a VNF-FGE algorithm which makes embedding decisions, according to the objective to optimize. Here, as in the VNE, there are virtual node and link mapping phases. In the

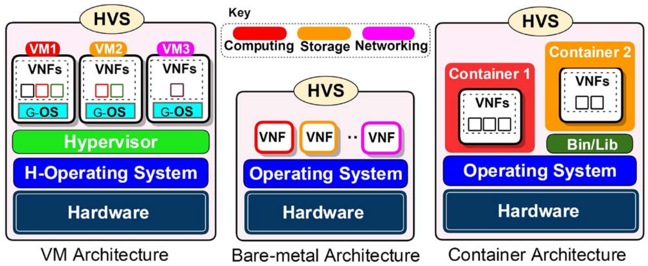

Fig. 7. VNFs deployment options.

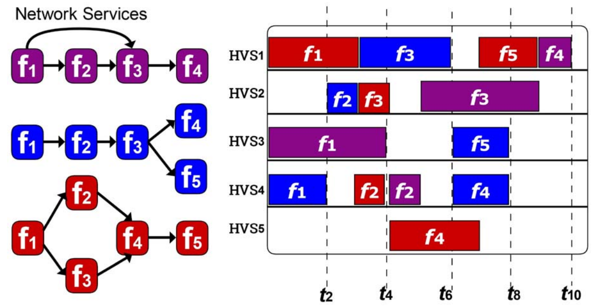

Fig. 8. VNFs Scheduling.

virtual node mapping phase; VNF 1 is hosted onto HVS 1, similarly, VNF 2 is embedded onto HVS 2, then, both VNF 3 and VNF 4 are mapped onto HVS 3 and finally VNF 5 is allocated onto HVS 4. In the virtual link mapping phase, the algorithm maps each virtual link between VNFs to a path in the SN. It is important to note that the path may be composed of more than one physical link; for instance, the virtual link between VNF 3 and VNF 4 is mapped into the path composed of the physical links HVS 3 - HVS 5 and HVS 5 - HVS 4.

To support the embedding stage, NFVI typically includes computing, storage and network hardware that provide processing, storage and connectivity for VNFs through the virtualization layer. In a hosted architecture VMs are allocated onto hypervisors which manage their network connectivity [61], [62]. While on bare-metal architectures, VNFs directly access to the server's physical resources, without using the hypervisor [63]. Another recent architecture to deploy VNFs is Container Virtualization [64]. This is a more efficient virtualized element than VMs that can run a variety of VNFs and others applications in the cloud. Fig. 7 illustrates the different architectures on which VNFs can be deployed.

The VNF-FGE stage may also be dynamic; it brings an additional dimension of complexity in terms of keeping track of where a given VNF is running. In other words, the orchestrator may trigger the migration of a VNF from a HVS to another if necessary, to rearrange VNFs of several services, in order to optimize the use and allocation of physical resources.

Most of existing NFV-RA approaches solve just the VNF-FGE stage; they consider the VNF-FG as a given input of the problem. A good example of a VNF-FGE approach is presented in [12]. Here, Bari *et al.* [12] propose to augment the SN in order to cope with the fact that virtual resources can be shared between multiple requests. With this augmented graph, they create an ILP aiming to minimize the OPEX caused by the allocation to the service provider while guaranteeing per service delay bounds. The OPEX is composed of: VNF deployment cost, energy cost and the cost of forwarding traffic. Due to scalability issues of the ILP, the authors present a heuristic that uses multi-stage graphs and the viterbi algorithm to solve the problem reaching near optimal performance results with a noticeable execution time reduction. Also, Elias *et al.* [65] formulate the centralized version of VNF-FGE as a non-linear integer optimization model where the objective function is to minimize the total network congestion. In addition, they present a game theory based approach to solve the problem in a fully distributed fashion resulting in near optimal solutions.

TABLE I
RUNTIME SERVICES

Service	VNFs	HVSs	Runtime (Time Units)
S.1	4	4	10
S.2	5	4	8
S.3	5	4	9

3) VNFs - Scheduling (VNFs-SCH): A third and final stage of the NFV-RA problem is the scheduling process, that we call Virtual Network Functions Scheduling. This stage attempts to give an answer to the following question: is it better to execute each function in order to minimize the total execution time without degrading the service performance and respecting all the precedences and dependencies between the VNFs composing the NS?

The NFV infrastructure is comprised of several and different HVSs, therefore, a proper scheduling of VNFs' execution should be performed in order to minimize the total execution time of the network services, and thus obtain improved performance.

Fig. 8 is an example that illustrates an example on how three different NSs, with different chains, and different network functions, can be scheduled over a limited NFVI, five servers in this case, minimizing the total execution time of the service set in order to maximize the system performance. Service 1 is composed of four VNFs; e.g., VNF 1 runs onto HVS 1 and takes 2.5 time units, VNF 2 runs onto HVS 4 and takes 1.0 time unit, VNF 3 runs on HVS 2 and takes 1.0 time unit, VNF 4 runs onto HVS5 and takes 2.5 time units, VNF 5 runs onto HVS 1 and takes 2.0 time units. The runtime of service 1 was 10 time units. While service 2 has the shortest runtime. Table I shows the summary information for the services 1, 2 and 3 respectively.

The effectiveness, performance, and efficiency of the scheduling process can be defined in terms of: *i*) number of available HVSs in the NFV infrastructure so they can process the functions composing the services; *ii*) the computing capacity of each server to process all the assigned functions; and *iii*) the complexity of the different network services, i.e., the number of functions composing each service.

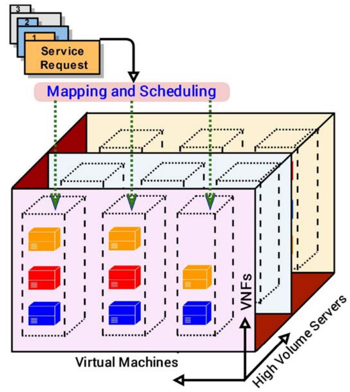

Fig. 9. NFV-RA Mapping and Scheduling.

Considering that NFV is still seen as a concept under investigation, little research has been conducted on VNFs scheduling. Riera *et al.* [66], [67] provided the first formalization of the scheduling problem in NFV as a Resource Constrained Project Scheduling Problem. Here, the final objective is to determine a feasible schedule with a minimal makespan (i.e., the final execution time of the last VNF of the last executed NS). However, although the problem is formulated, no solution is proposed to solve the problem in [66] and [67]. Recently, Mijumbi *et al.* [68] proposed an approach to tackle the online VNF-FGE and VNFs-SCH by proposing greedy and metaheuristic (tabu search) approaches aiming at reducing the flow execution time. The algorithms perform both mapping and scheduling at the same time (one-shot) resulting in high acceptance ratio, low average flow time and low embedding cost. This work considers a resource sharing approach that allows a given VM to process multiple VNFs, one after another (possibly) from a queue, as is illustrated in Fig. 9.

Automation is paramount to the success of NFV. This calls for, among other things, algorithms that are able to perform the composition, mapping and scheduling of VNFs in an online and dynamic fashion. These algorithms must ensure that physical hardware resources are used efficiently. The success of NFV will depend, in part, on the existence and performance of algorithms that determine how, where, and when the VNFs are instantiated and executed.

IV. NFV-RA Problem Features

In this section, we discuss issues related to parameters, embedding objectives, coordination and optimization strategies that can be used by NFV-RA approaches. Besides, we include various useful metrics to assess the performance of NFV-RA solution proposals and existing simulation tools to implement and evaluate such algorithm proposals.

A. NFV-RA Parameters

Resources in the NFV-RA problem are annotated with parameters. Physical resources have individual capacities and types. For example, a substrate node can provide a certain computing capacity relating to the CPU available to it. In contrast, a virtual node will require a certain computing capacity in order to properly compute routing information. These parameters are of paramount importance in order to achieve a valid embedding stage. Parameters can be categorized according to several dimensions.

As a first step, one can distinguish between node and link parameters. Node parameters are attributes that refer to nodes. Depending of the type of function a physical node is able to host, its parameters may differ; for instance, a storage node has big memory resources while a processing node counts on big CPU resources. Link parameters are attributes that refer to links, such as bandwidth or propagation delay. Fischer *et al*. [69], have presented a categorization of parameters for VNE algorithms, that could be also used in the NFV-RA scenario. For instance, Mehraghdam *et al*. [56] have considered parameters such as latency, computational resources, data rate capacity, and data rate demand for the placement of VNF chains. Also, another important parameters such as CPU and bandwidth, were considered in [52] and [70]. Basta *et al.* [24] proposed a model that resolves the functions placement and aims at minimizing the transport network load overhead against several parameters such as data-plane delay, number and placement of potential datacenters and control overhead.

B. Coordination of NFV-RA Stages

The NFV-RA problem is completely solved when its three stages VNFs-CC, VNF-FGE and VNFs-SCH, are solved. Most of the existing approaches focus their attention on solving one of the problem stages (in particular, the VNE-FGE stage). In this context, some approaches solve stages VNFs-CC, VNF-FGE [71] or VNFs-SCH [66] independently. However, lastly, some approaches have been proposed to solve more than one stage of the NFV-RA problem. These approaches can act in an *uncoordinated* or a *coordinated* fashion:

- *Uncoordinated*: As it has been described in Section III-C, the output of each NFV-RA stage is the input of the one that follows. However, some approaches attempting to solve more than one stage do not coordinate them in order to obtain a better solution. A good example that demonstrates this lack of coordination was proposed in [56] to solve the VNFs-CC and VNF-FGE stages of the problem where: 1) a greedy heuristic is proposed to generate a VNF-FG trying to reduce total data rate of the flows and, 2) once the service request has been chained, the VNF-FGE sub-problem is solved exactly by using a mixed integer quadratic constrained program with regard to three different optimization criteria: a) maximizing remaining data rate on network links, b) increasing energy efficiency, and c) minimizing the latency of the assigned substrate paths. As discussed in this paper, solving VNF-FGE with an exact algorithm comes with high cost in terms of runtime for big networks, making it unsuitable for medium to large-size scenarios. Also, solving the problem in separated stages does not guarantee that the generated VNF-FG is able to be embedded in the second stage, even if there is a feasible solution for the whole problem.

- *Coordinated*: In contrast, coordinated approaches try to perform each stage in a way that its result is prepared to optimize the following stage. In fact, some approaches try to perform more than one NFV-RA's stages in one step. A good example is presented in [52]. The approach introduces the following contributions: 1) *coordinated allocation* as it proposes a recursive heuristic that solves VNFs-CC and VNF-FGE in a coordinated way, that is, the VNF-FG is composed and allocated simultaneously. In this way, the proposed approach improves the likelihood of successful embedding as the building of the VNF-FG depends, in each step, on its successful allocation and 2) *scalability* as the heuristic nature of the proposal computes results within reasonable runtimes with negligible performance effects when compared against [56]. Another coordinated approach is presented in [68], where VNF-FGE and VNFs-SCH are heuristically solved in a coordinated way. The approach introduces the time to execute each VNF for a specific service in order to include a constraint that forbids a physical node to execute more than one VNF at a certain time as part of the embedding phase. The main objective of the proposal is to minimize the total flow time, i.e., the final execution time of the last VNF of the last executed NS. The paper proposes a greedy heuristic and a tabu search metaheuristic to solve the problem with very good results in terms of acceptance ratio and flow time.

C. NFV-RA Optimization Strategies

The NFV-RA is a $\mathcal{NP}$-hard optimization problem, as it can be seen as a generalization of the VNE which is, in turn $\mathcal{NP}$-hard [47], meaning that runtimes to optimally solve the problem are unaffordable for medium to large instance sizes. This kind of problems can be solved by applying the following optimization strategies: *exact*, *heuristic* or *metaheuristic*.

Exact solutions propose optimal techniques to solve small instances of the problem and to create baseline solutions that represent an optimal bound for heuristic-based solutions. Heuristic-based solutions are not expected to always find the global optimum. Instead, they try to find a good solution while keeping execution time low. Usually, heuristic solutions suffer from the problem that they can get stuck in a local optimum that can be far away from the real optimum. Metaheuristic solutions improve the quality of the result by escaping from local optima in reasonable and tunable running times.

i) Exact solutions: Optimal solutions can be achieved by means of linear programming algorithms. Particularly, ILP can be used to optimally formulate the VNF-FGE problem. Although ILPs are in many practical situations $\mathcal{NP}$-complete, there are exact algorithms (e.g., branch and bound or branch and price) that solve small instances of the problem in reasonable time [72]. Software tools implementing these algorithms, commonly called solvers, are available either as open-source (e.g., GLPK [73]), or proprietary (e.g., CPLEX [74]).

A good example of exact strategies is presented in [75]. Here, the authors consider a NFV-enabled hybrid environment when specific and general purpose hardware coexist. Their proposal is to solve the VNE-FGE problem by formulating an ILP where the objective is to minimize the number of used physical nodes. Due to the inherent ILP complexity, the performance of the proposal is evaluated in a small service provider scenario under different traffic loads. Another exact strategy is presented in [76] where a network-enabled cloud is considered the NFVI. An ILP is formulated there to solve VNF-FGE where the objective is to minimize the used bandwidth in the physical network. Results show that when all the physical nodes are NFV-enabled, the bandwidth savings are greater than in hybrid physical network scenarios.

ii) Heuristic solutions: Execution time is crucial in NFV-RA. NFV deals with dynamic online environments where arrival time of service requests is not known in advance. Therefore, to avoid delay when solving the NFV-RA problem, the execution time of the proposed algorithms should be minimized. Accordingly, heuristic-based solutions are proposed.

An example of a heuristic approach to solve the VNF-FGE stage in WLANs NFV scenarios is presented in [77] where a recursive greedy algorithm is implemented trying to map VNFs to physical node trying to balance the total network load. For link mapping, the approach uses shortest path algorithms. Another heuristic approach to solve VNE-FGE trying to minimize the OPEX is presented in [71]. In this approach, the authors propose to separate VNF-FGE in two well-known $\mathcal{NP}$-hard problems: the facility location problem and the generalized assignment problem. Then, they introduce rounding-based heuristics to solve the problem and obtain better results with regard to OPEX when compared with greedy-based solutions.

iii) Metaheuristic solutions: NFV-RA can be seen as a combinatorial optimization problem where an optimal solution is sought over a discrete search-space. As the optimal solution for large instances of these problems is hard to find, metaheuristics can be used to find near-optimal solutions by iteratively improve problem solutions with regard to a given measure of quality [78].

To the best of our knowledge, the only metaheuristic-based NFV-RA approach is presented in [68]. Here, tabu search is used to solve VNF-FGE and VNFs-SCH in one step aiming at reducing the flow execution time. Simulation results show high acceptance ratio, low average flow execution time and low embedding cost.

D. NFV-RA Objectives and Metrics

Solving NFV-RA requires a strategic objective to optimize. Objectives related with QoS, profit maximization, fault tolerance, load balancing, energy saving, etc., are common examples of NFV-RA's main goals.

There are several situations where these requirements are explicit in the request. For instance, a network service that provides voice over IP services needs to count on medium bandwidth and low delay requirements. From TSP's point of view, a natural objective would be to maximize the economic benefit of accepting service requests (long-term average revenue); this objective is directly proportional to maximize the number of allocated service requests (acceptance ratio). In order to reach this goal, NFV-RA approaches should try to minimize the resources spent by the NFVI (cost minimization). Resilience in NFV-RA may also be brought into the picture by integrating fallback resources within the NFVI. Backup nodes/links can be setup either for all or just for some specific nodes/links that may fail. Besides, multi-objective functions (i.e., minimum CPU utilization and minimum energy consumed) could be considered.

Different optimization objectives have been proposed in several NFV-RA approaches. For instance, the objective function of [70] aims at minimizing the number of virtual network function instances mapped on top of the NFVI. Another example is presented in [79], where the objective of the algorithm is to maximize the number of successfully embedded service function chaining requests. Mehraghdam *et al*. [56] try to solve the stage 2 of NFV-RA by considering three different objectives: 1) maximizing remaining data rate, 2) minimizing the number of used nodes and 3) minimizing the latency of created paths.

Another important aspect are the NFV-RA problem metrics. Metrics are necessary to assess the quality of a NFV-RA solution proposal. They are used to compare different approaches and to quantify advances in optimization. Different metrics can be structured according to the objective function as discussed above.

Below, we mention some performance metrics used for measuring and evaluating the VNF placement algorithm of the NFV-RA problem.

- Mijumbi *et al*. [68] carry out evaluations of RA algorithms considering metrics such as *successful service mappings*, *total service processing times*, *revenue* or *cost*, under varying network conditions.
- Important metrics such as *number of active physical nodes*, *node buffer capacity*, *function processing times* and *function buffer demand* were also considered in [68].
- Additional metrics such as *acceptance rate*, *resources utilization* and *traffic congestion*, were used in [80]–[82].

E. Simulation Tools

In order to evaluate NFV-RA proposals, simulation tools are needed. Typically, algorithms are run with a randomly generated set of scenarios. Each of these scenarios consists of a NFVI and a set of VNFRs to be attended. Appropriate parameters are assigned to both substrate and virtual resources. After the algorithms have tried to attend the VNFRs, results are evaluated using one or more metrics. The following are some simulation tools that could be used in a NFV-RA ecosystem.

Some simulation environments have been developed to work with NFV-RA-oriented algorithms. However, they are closed solutions for specific proposals that difficult the possibility of comparing different approaches [24].

An example of a wider simulation environment is ALEVIN [83], a Java project with well-defined interfaces to implement new optimization algorithms and metrics. While ALEVIN was created to implement solutions to the VNE problem; Beck and Botero [52] have made some modifications to the code to adapt it to the requirements of NFV-RA.

F. Taxonomy of NFV Approaches

Here, a taxonomy of the main proposals to solve NFV-RA is presented. Table II summarizes the different NFV-RA approaches with regard to their main characteristics. Each work is classified according to 1) the **stage or set of stages** it solves (VNFs-CC, VNFs-FGE or VNFs-SCH) and the coordination strategy when it solves more than one stage, 2) solution **strategy** (exact, heuristic or metaheuristic), 3) the **scenario** of application of each solution is identified; and finally 4) the main **contribution** of each approach is summarized.

V. Emerging Research Challenges

NFV is still in the early stages. There are still important aspects that should be investigated to efficiently manage and allocate the use of the resources in NFV-based network architectures. This section discusses future research directions that we consider of paramount importance for the development and implementation of the NFV technology.

A. NFV-RA Coordination Stages

Solving the resource allocation problem in NFV-based network architectures in a coordinated way is a major challenge. The NFV-RA has three stages that are related to each other, and the intention is to carry out the execution of each stage in a coordinated way. The aim is to optimize the use of resources which would improve the performance of the network. It will also facilitate the migration VNFs of a HVS another nimbly. To the best of our knowledge, there are not proposals that try to solve the three stages of NFV-RA in a coordinated way.

B. Dynamic NFV-RA

In real situations, NFV-RA has to be tackled as an online problem. That is, service requests (VNFRs) are not known in advance. Instead, they arrive to the system dynamically and stay in the network an arbitrary time. Therefore, a NFV-RA algorithm must handle the VNFRs as they arrive, rather than attending a set of VNFRs at once. While in principle, all approaches can be operated in an online manner, static NFV-RA approaches do not contemplate the possibility of remapping, or even recomposing, one of more VNFRs to improve the allocation performance. Also, it is worth mentioning that VNFRs may change over time, that is, new VNFs may be added or deleted from a VNFR meaning that recomposition, remapping and rescheduling may be necessary.

C. Resources Discovery

With VNFs executing on shared environments, the qualification of resources available on NFVI should be measurable regarding to computing, storage and network domains

TABLE II
SUMMARY OF STATE-OF-THE-ART NFV-RA APPROACHES

References	Scenario	Stage	Strategy	Contribution
Bari et al. [12] (2015)	TSP's networks	VNF-FGE	Exact, Heuristic	Provides an ILP formulation with implementation in CPLEX and a dynamic programming based heuristic to solve larger instances of the VNFs-FGE problem.
Mijumbi et al. [68] (2015)	TSP's networks	VNF-FGE, VNFs-SCH (Coord)	Heuristic, meta-heuristic	Formulates the online virtual function mapping and scheduling problem and proposes a set of heuristic and metaheuristic algorithms to solve the VNF-FGE and VNFs-SCH in a coordinated way.
Basta et al. [24] (2014)	Mobile network	VNF-FGE	Heuristic	An optimization model is presented for the placement of mobile core gateways (SGWs and PGWs) with respect to latency constraints and different gateway virtualization scenarios.
Beck and Botero [52] (2015)	TSP's networks	VNFs-CC, VNF-FGE (Coord)	Heuristic	Proposes a heuristic method to coordinate the composition of VNF chains and their embedding into the substrate network.
Mehraghdam et al. [56] (2014)	TSP's networks	VNFs-CC, VNF-FGE (Uncoord)	Exact, Heuristic	Provide a model for formalizing the chaining of NFs using a context-free language and describes the VNE-FGE as a mixed integer quadratically constrained program.
Luizelli et al. [70] (2015)	NFV networks	VNF-FGE	Exact, Heuristic	Formalizes the network function placement and proposes an ILP model trying to minimize the number of deployed VNF instances. To cope with ILP's complexity, it provides a heuristic to solve larger problem sizes.
Moens et al. [75] (2014)	Hybrid networks	VNF-FGE	Exact	Presents and evaluates a formal model trying to to minimize the number of used HVSs within hybrid NFV environments.
Addis et al. [62] (2015)	TSP's networks	VNF-FGE	Exact	The authors solve the problem of VNF service chain placement using a mixed ILP and give insights into trade-offs between legacy and NFV-based Traffic Engineering.
Riera et al. [66] (2014)	Optical networks	VNFs-SCH	Exact	Provides a formalization model of the VNFs-SCH, i.e. finding the corresponding time slots for functions to be executed over a given set of machines.
Riera et al. [67] (2014)	NFV networks	VNFs-SCH	—	Provides the first formalization model for the VNF complex scheduling problem, using the complex job formalization.
Clayman et al. [84] (2014)	LAN/DC	VNFs-CC	Exact	Specifies an architecture based on an orchestrator that ensures the automatic placement of the virtual nodes and the allocation of NSs on them.
Xia et al. [85] (2015)	Packet/Optical DC	VNF-FGE	Heuristic	Formulates the problem of optimal VNF placement in binary integer programming placement in packet/optical data centers, such that the overall optical to electronic to optical conversions can be minimized, and propose an alternative efficient heuristic algorithm to solve it.
Riggio et al. [79] (2015)	Wireless networks	VNF-FGE	Exact	Formalizes the VNF placement problem for radio access networks as an ILP problem. Also it proposes a VNF placement heuristic to solve it.
Riera et al. [80] (2015)	TSP's networks	VNFs-SCH	—	An analytic model for the VNF-FG is proposed with the aim to optimize the execution time of the deployed network services.
Ghaznavi et al. [81] (2015)	Cloud networks	VNF-FGE	Heuristic	Introduces the elastic virtual network function placement problem and presents a model for minimizing operational costs in providing VNF service.
Bruschi et al. [82] (2016)	TSP's networks	VNF-FGE	Heuristic	Presents and evaluates an energy-aware game theory-based solution for resource allocation of VNFs within NFV environments.
Martini et al. [86] (2015)	5G networks	VNF-FGE	Exact	Formulates the problem of composing and computing of VNFs to select nodes along the path that minimize the overall latency.
Elias et al. [65] (2015)	TSP's networks	VNF-FGE	Exact, Heuristic	Formulates the network functions composition problem as a non-linear optimization model to accurately capture the congestion of physical resources.
Baumgartner et al. [87] (2015)	Mobile networks	VNF-FGE	Exact	Proposes a novel ILP formulation which combines the optimization of the virtual network topology with VNE optimization.
Mohammadkhan et al. [88] (2015)	TSP's networks	VNF-FGE	Heuristic	Provides a mixed ILP formulation to determine the placement of services and routing of the flows; and heuristics to provide the opportunity to perform the placement incrementally without imposing a significant penalty.
Soares and Sargento [89] (2015)	Cloud networks	VNF-FGE	Exact	An optimal formulation and an embedding strategy based on ILP are presented. The proposed strategy takes into account the load balancing of DC and inter-DC network domains.
Gupta et al. [76] (2015)	TSP's networks	VNFs-CC, VNF-FGE (Uncoord)	Exact	Presents a mathematical model for the placement of VNFs which ensures the service chaining required by traffic flows.
Lukovszki and Schmid [90] (2015)	LAN/DC	VNFs-CC, VNF-FGE (Uncoord)	Exact	The main contribution is a deterministic competitive online algorithm called ACE (Admission control and Chain Embedding)
Mijumbi et al. [91] (2015)	Mobile networks	VNF-FGE	Exact, Heuristic	Defines the virtualized radio access network placement and assignment problem and formulates it as a binary integer linear program. Also, proposes a greedy approximation to solve larger instances.
Riggio et al. [77] (2015)	NFV networks	VNF-FGE	Heuristic	Proposes an algorithm for virtual network function placement which optimizes the functions deployment according to application level constraints.
Bagaa et al. [92] (2014)	Cloud networks	VNF-FGE	Heuristic	Proposes three heuristics to solve the placement problem of mobile network functions over federated cloud.
Bellavista et al. [93] (2015)	Cloud networks	VNF-FGE	Exact, Heuristic	Introduces and discusses some challenging technical issues associated with optimal placement problem of entire VDCs, taking into account multiple virtual and physical resources and constraints.
Lin et al. [94] (2015)	TSP's networks	VNF-FGE	Exact	Evaluates the performance of the virtual network functions placement in terms of its ability to support end-to-end requests with limited physical resources.
Sahhaf et al. [95] (2015)	Carrier networks	VNFs-CC, VNF-FGE (Uncoord)	—	Details the role of a scalable orchestrator in charge of finding and reserving adequate resource.
Németh et al. [96] (2015)	Carrier networks	VNF-FGE	Heuristic	Demonstrates novel approaches to design, evaluate and fine-tune real-time parameterizable orchestration algorithms for carrier grade networks.
Sahhaf et al. [97] (2015)	NFV networks	VNFs-CC, VNF-FGE (Uncoord)	Exact, Heuristic	Proposes two novel algorithms to map service function chains to the network infrastructure while allowing possible decomposition of network functions. The first algorithm is based on ILP, and the second one is a heuristic algorithm.
Bouet et al. [98] (2015)	TSP's networks	VNF-FGE	Heuristic	Formulates the virtual deep packet inspection placement problem as a cost minimization problem. Besides, formulates the problem as a multi-commodity flow problem and solve it as an ILP.
Bauschert et al. [99] (2015)	Mobile network	VNF-FGE	Exact	Presents a novel mathematical optimization model for virtual mobile core network embeddings with respect to latency bounds.

TABLE II (continued)

Cohen et al. [71] (2015)	TSP's networks	VNF-FGE	Heuristic	Proposes rounding-based heuristics to solve VNF-FGE trying to minimize TSP's OPEX.
Lin et al. [100] (2016)	Optical networks	VNF-FGE	Exact, heuristic	Proposes a mixed integer linear program trying to minimize the costs of VNFs' deployment and traffic routing. It also proposes a heuristic algorithm based on game theory.
Riggio et al. [101] (2016)	Wireless networks	VNF-FGE	Exact, Heuristic	Formalizes the wireless VNF placement problem as an ILP trying to compute the optimal VNF placement based on the available radio resources. Besides, it proposes a greedy heuristic named to solve the problem.
Ghaznavi et al. [102] (2016)	Data center networks	VNF-FGE	Exact, Heuristic	Introduces a mixed integer programming problem trying to minimize the host and bandwidth deployment costs, and proposes a scalable heuristic to solve large instances of the problem.
Zhang et al. [103] (2016)	TSP's networks	VNF-FGE	Heuristic	Introduces a vertex centric based distributed approach to perform the VNF-FGE in multi-domain networks.
Jang et al. [104] (2016)	TSP's networks	VNF-FGE	Exact	Proposes a Linear Programming-based solution to solve the VNF-FGE problem trying to minimize the network resource usage.
Kuo et al. [105] (2016)	Data Center Networks	VNF-FGE	Heuristic	Proposes a chain deployment algorithm to find a solution considering the tradeoff between path length and VM reuse factor.

in automatic/on-demand ways. Discovering NFV infrastructure idle resources incurs on evolved cloud computing methods, e.g., scheduling of VNFs requires bandwidth and latency metrics collected through distributed measurement mechanisms.

D. QoS Compliant Allocations

Network services must be deployed in the NFVI so that the QoS parameters negotiated between the TSP and the end user in the SLA, are guaranteed. Up to now, most of the existing NFV-RA proposals guarantee bandwidth between VNFs, computation power and memory in physical nodes and they even consider end to end latency [12], [56], [68], [70]. Experimental benchmarks to perform resource monitoring and enable runtime resource evaluation and test-before-deploy have been recently proposed [106]. Also, validation and testing of NFV management and orchestration of the QoS compliance is an important branch of research [107].

However, QoS parameters such as jitter (for real time services) or loss probability (for availability related services) have not been considered up to now. Also, as it is detailed next, survivability has not been still tackled in NFV-RA.

E. Chain Re-Composition and Resilience to Failure

The success of a network service directly depends on the high levels of availability and reliability of the hardware and software. In the NFV landscape, network service requirements should provide resilience to failure, service continuity, and service assurance. Resilience to failure is provided by implementing an automated on-demand mechanism in the NFV framework to reconstitute the chain of VNFs after a failure. Chain recomposition should not have any impact on the system to ensure stable service. Service assurance is provided by the NFV orchestrator, which is monitoring network-function performance and scale resources almost in real time [54].

F. Energy Saving

In terms energy efficiency, the current trend is towards renewable energy and eco-friendly solutions, there is a need to consider the type of energy sources used in the NFVI as a parameter in the NFV-RA process (placement, scheduling and chaining algorithms). The implementation of NFV foresees to reduce OPEX including the levels of energy consumption. We hope that the storage and processing of VNFs on the cloud can significantly reduce energy consumption. Future research could be focused on the provision of energy-aware strategies to find efficient NFV-RA solutions within reasonable time.

It is important also to avoid rejecting VNFRs when shifting to more energy efficient NFV-RA solutions. Besides, topology and load dependence are also non-negligible aspects in energy consumption. Recent work on the energy possibilities of NFV is presented in [108].

VI. Conclusion

The wide deployment of future network architectures based on NFV will depend largely on the success of resource allocation. This challenge has been called the NFV-RA problem. It is comprised of three different stages, the first stage is the VNFs-CC, which seeks to concatenate the VNFs efficiently in order to compose an NS in the most adequate way, with respect to the service provider's goals. The second stage is the VNF-FGE which seeks to find where in the NFVI, the VNFs will be allocated in a suitable way, considering the requirements of individual requests as well as the overall requirements of all NSs. The third and final stage is the VNFs-SCH, which seeks to determine when is better to execute each function into the NFVI in order to minimize the total execution time without degrading the service performance and respecting all the precedences and dependencies between the VNFs composing the service at the same time.

This paper has discussed a comprehensive survey about NFV-RA problem. Our key contribution is to give a detailed categorization of research works in the NFV-RA context, including works-oriented to chains composition, embedding, and scheduling of VNFs into NFVI. Finally, we summarized the main research challenges appearing in the NFV-RA realm.

References

[1] *C-RAN: The Road Towards Green RAN; White Paper. Version 2.5*, China Mobile Res. Inst., Beijing, China, Oct. 2011.

[2] J. Wu, Z. Zhang, Y. Hong, and Y. Wen, "Cloud radio access network (C-RAN): A primer," *IEEE Netw.*, vol. 29, no. 1, pp. 35–41, Jan./Feb. 2015.

[3] J. Sherry and S. Ratnasamy, "A survey of enterprise middlebox deployments," Dept. EECS, Univ. California, Berkeley, CA, USA, Tech. Rep. UCB/EECS-2012-24, Feb. 2012.

[4] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *IEEE Commun. Mag.*, vol. 53, no. 2, pp. 90–97, Feb. 2015.

[5] M. Chiosi *et al.*, "Network functions virtualisation: An introduction, benefits, enablers, challenges and call for action," in *Proc. SDN OpenFlow World Congr.*, Darmstadt, Germany, Oct. 2012, pp. 22–24.

[6] ETSI. European Telecommunications Standards Institute. Accessed on Feb. 28, 2016. [Online]. Available: http://www.etsi.org/technologies-clusters/technologies/nfv

[7] B. Hirschman *et al.*, "High-performance evolved packet core signaling and bearer processing on general-purpose processors," *IEEE Netw.*, vol. 29, no. 3, pp. 6–14, May/Jun. 2015.

[8] Z. Bronstein, E. Roch, J. Xia, and A. Molkho, "Uniform handling and abstraction of NFV hardware accelerators," *IEEE Netw.*, vol. 29, no. 3, pp. 22–29, May/Jun. 2015.

[9] *Network Functions Virtualisation (NFV): Architectural Framework*, ETSI Ind. Spec. Group (ISG), Valbonne, France, Dec. 2014.

[10] R. Mijumbi *et al.*, "Network function virtualization: State-of-the-art and research challenges," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 236–262, 1st Quart. 2016.

[11] A. Gember*et al.*, "Stratos: A network-aware orchestration layer for middleboxes in the cloud," *CoRR*, vol. abs/1305.0209, 2013. [Online]. Available: http://arxiv.org/abs/1305.0209

[12] M. F. Bari, S. R. Chowdhury, R. Ahmed, and R. Boutaba, "On orchestrating virtual network functions," in *Proc. 11th Int. Conf. Netw. Service Manag. (CNSM)*, Barcelona, Spain, Nov. 2015, pp. 50–56.

[13] ETSI ISG NFV (Operator Group), "Network functions virtualisation—Network operator perspectives on industry progress," White Paper, Oct. 2013.

[14] J. de Jesus Gil Herrera and J. F. B. Vega, "Network functions virtualization: A survey," *IEEE Latin America Trans.*, vol. 14, no. 2, pp. 983–997, Feb. 2016.

[15] ETSI Industry Specification Group (ISG). (Aug. 2015). *Draft Documents*. [Online]. Available: https://docbox.etsi.org/isg/nfv/open/Drafts/

[16] C. Price and S. Rivera, "Opnfv: An open platform to accelerate NFV," White Paper, 2012.

[17] C. Cui, "Network functions virtualisation: Network operator perspectives on industry progress," in *Proc. SDN OpenFlow World Congr.*, Düsseldorf, Germany, White Paper no. 3, no. 1, Oct. 2014.

[18] *Network Functions Virtualisation: Use Cases*, ETSI Ind. Specification Group (ISG), Valbonne, France, Oct. 2013.

[19] W. Rankothge, J. Ma, F. Le, A. Russo, and J. Lobo, "Towards making network function virtualization a cloud computing service," in *Proc. IFIP/IEEE IM*, Ottawa, ON, Canada, May 2015, pp. 89–97.

[20] J. F. Monserrat *et al.*, "Rethinking the mobile and wireless network architecture: The METIS research into 5G," in *Proc. Eur. Conf. Netw. Commun. (EuCNC)*, Bologna, Italy, Jun. 2014, pp. 1–5.

[21] G. Carella *et al.*, "Cloudified IP multimedia subsystem (IMS) for network function virtualization (NFV)-based architectures," in *Proc. IEEE ISCC*, Funchal, Portugal, Jun. 2014, pp. 1–6.

[22] M. Peng, Y. Li, Z. Zhao, and C. Wang, "System architecture and key technologies for 5G heterogeneous cloud radio access networks," *IEEE Netw.*, vol. 29, no. 2, pp. 6–14, Mar./Apr. 2015.

[23] I. F. Akyildiz, S.-C. Lin, and P. Wang, "Wireless software-defined networks (W-SDNS) and network function virtualization (NFV) for 5G cellular systems: An overview and qualitative evaluation," *Comput. Netw.*, vol. 93, pp. 66–79, Dec. 2015.

[24] 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 *Proc. 4th Workshop All Things Cellular Oper. Appl. Challenges (AllThingsCellular)*, Chicago, IL, USA, 2014, pp. 33–38.

[25] Z. Bronstein and E. Shraga, "NFV virtualisation of the home environment," in *Proc. IEEE 11th Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, Jan. 2014, pp. 899–904.

[26] 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, Oct. 2014, pp. 344–349.

[27] 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, Oct. 2014, pp. 832–833.

[28] S. Pachnicke, A. Magee, P. Turnbull, K. Grobe, and J.-P. Elbers, "Functional and structural convergence of fixed/mobile access networks," in *Proc. ITG Symp. Photon. Netw.*, Leipzig, Germany, May 2015, pp. 1–5.

[29] J. Montalvo, M. Arroyo, J. A. Torrijos, J. Lorca, and I. Berberana, "Fixed-mobile convergence and virtualization in 5G optical transport networks," in *Proc. 17th Int. Conf. Transp. Opt. Netw. (ICTON)*, Budapest, Hungary, Jul. 2015, pp. 1–4.

[30] R. Szabo *et al.*, "Elastic network functions: Opportunities and challenges," *IEEE Netw.*, vol. 29, no. 3, pp. 15–21, May/Jun. 2015.

[31] *Network Functions Virtualisation (NFV); Management and Orchestration*, ETSI Ind. Specification Group (ISG), Valbonne, France, Dec. 2014.

[32] D. Kreutz *et al.*, "Software-defined networking: A comprehensive survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, Jan. 2015.

[33] T. Wood, K. K. Ramakrishnan, J. Hwang, G. Liu, and W. Zhang, "Toward a software-based network: Integrating software defined networking and network function virtualization," *IEEE Netw.*, vol. 29, no. 3, pp. 36–41, May/Jun. 2015.

[34] S. Sun, M. Kadoch, L. Gong, and B. Rong, "Integrating network function virtualization with SDR and SDN for 4G/5G networks," *IEEE Netw.*, vol. 29, no. 3, pp. 54–59, May/Jun. 2015.

[35] Y. Li and M. Chen, "Software-defined network function virtualization: A survey," *IEEE Access*, vol. 3, pp. 2542–2553, 2015.

[36] E. Hernandez-Valencia, S. Izzo, and B. Polonsky, "How will NFV/SDN transform service provider opex?" *IEEE Netw.*, vol. 29, no. 3, pp. 60–67, May/Jun. 2015.

[37] P. Quinn and J. Guichard, "Service function chaining: Creating a service plane via network service headers," *Computer*, vol. 47, no. 11, pp. 38–44, Nov. 2014.

[38] IETF. (2015). *The Internet Engineering Task Force Service Function Chaining Working Group. Documents*. [Online]. Available: https://datatracker.ietf.org/wg/sfc/documents/

[39] P. Quinn and T. Nadeau, "Problem statement for service function chaining," Internet Requests for Comments, IETF, Fremont, CA, USA, RFC 7498, Apr. 2015. [Online]. Available: http://www.rfc-editor.org/rfc/rfc7498.txt

[40] J. Halpern and C. Pignataro, "Service function chaining (SFC) architecture," Internet Requests for Comments, RFC Editor, RFC 7665, Oct. 2015. [Online]. Available: http://www.rfc-editor.org/rfc/rfc7665.txt

[41] A. Belbekkouche, M. M. Hasan, and A. Karmouch, "Resource discovery and allocation in network virtualization," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 4, pp. 1114–1128, 4th Quart. 2012.

[42] A. Fischer, J.-F. Botero, M. T. Beck, H. de Meer, and X. Hesselbach, "Virtual network embedding: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 1888–1906, 4th Quart. 2013.

[43] J. F. Botero, M. Molina, X. Hesselbach-Serra, and J. R. Amazonas, "A novel paths algebra-based strategy to flexibly solve the link mapping stage of VNE problems," *J. Netw. Comput. Appl.*, vol. 36, no. 6, pp. 1735–1752, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804513000672

[44] B. Wang, X. Chang, J. Liu, and J. K. Muppala, "Reducing power consumption in embedding virtual infrastructures," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Anaheim, CA, USA, Dec. 2012, pp. 714–718.

[45] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking virtual network embedding: Substrate support for path splitting and migration," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 17–29, 2008.

[46] Y. Chen, J. Li, T. Wo, C. Hu, and W. Liu, "Resilient virtual network service provision in network virtualization environments," in *Proc. IEEE 16th Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Shanghai, China, Dec. 2010, pp. 51–58.

[47] E. Amaldi, S. Coniglio, A. M. C. A. Koster, and M. Tieves, "On the computational complexity of the virtual network embedding problem," *Electron. Notes Discr. Math.*, vol. 52, pp. 213–220, Jun. 2016.

[48] J. F. Botero *et al.*, "Energy efficient virtual network embedding," *IEEE Commun. Lett.*, vol. 16, no. 5, pp. 756–759, May 2012.

[49] J. F. Botero and X. Hesselbach, "Greener networking in a network virtualization environment," *Comput. Netw.*, vol. 57, no. 9, pp. 2021–2039, 2013.

[50] S. Liu, Z. Cai, H. Xu, and M. Xu, "Towards security-aware virtual network embedding," *Comput. Netw.*, vol. 91, pp. 151–163, Nov. 2015.

[51] S. Liu, Z. Cai, H. Xu, and M. Xu, "Security-aware virtual network embedding," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Sydney, NSW, Australia, Jun. 2014, pp. 834–840.

[52] M. T. Beck and J. F. Botero, "Coordinated allocation of service function chains," in *Proc. IEEE Glob. Commun. Conf. Sel. Areas Commun. Softw. Defined Netw. Netw. Functions (GC-SAC-SDN)*, San Diego, CA, USA, Dec. 2015, pp. 1–6.

[53] *Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV*, ETSI Ind. Specification Group (ISG), Valbonne, France, Dec. 2013.

[54] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "NFV: State of the art, challenges, and implementation in next generation mobile networks (vEPC)," *IEEE Netw.*, vol. 28, no. 6, pp. 18–26, Nov./Dec. 2014.

[55] W. John *et al.*, "Research directions in network service chaining," in *Proc. IEEE SDN Future Netw. Services (SDN4FNS)*, Trento, Italy, Nov. 2013, pp. 1–7.

[56] S. Mehraghdam, M. Keller, and H. Karl, "Specifying and placing chains of virtual network functions," in *Proc. IEEE 3rd Int. Conf. Cloud Netw. (CloudNet)*, Oct. 2014, pp. 7–13.

[57] Z. Kong, C.-Z. Xu, and M. Guo, "Mechanism design for stochastic virtual resource allocation in non-cooperative cloud systems," in *Proc. IEEE Int. Conf. Cloud Comput.*, Washington, DC, USA, Jul. 2011, pp. 614–621.

[58] M. F. Bari *et al.*, "Data center network virtualization: A survey," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 2, pp. 909–928, 2nd Quart. 2013.

[59] E. S. Correa, L. A. Fletscher, and J. F. Botero, "Virtual data center embedding: A survey," *IEEE Latin Amer. Trans.*, vol. 13, no. 5, pp. 1661–1670, May 2015.

[60] J. Soares *et al.*, "Toward a telco cloud environment for service functions," *IEEE Commun. Mag.*, vol. 53, no. 2, pp. 98–106, Feb. 2015.

[61] *Network Functions Virtualisation (NFV): Infrastructure; Hypervisor Domain*, ETSI Ind. Specification Group (ISG), Valbonne, France, Jan. 2015.

[62] B. Addis, D. Belabed, M. Bouet, and S. Secci, "Virtual network functions placement and routing optimization," in *Proc. IEEE Int. Conf. Cloud Netw.*, Niagara Falls, ON, Canada, Oct. 2015, pp. 171–177.

[63] P. Veitch, M. J. McGrath, and V. Bayon, "An instrumentation and analytics framework for optimal and robust NFV deployment," *IEEE Commun. Mag.*, vol. 53, no. 2, pp. 126–133, Feb. 2015.

[64] R. Rosen, *Resource management: Linux kernel namespaces and cgroups*, Haifux, Haifa, Israel, May 2013.

[65] J. Elias, F. Martignon, S. Paris, and J. Wang, "Efficient orchestration mechanisms for congestion mitigation in NFV: Models and algorithms," *IEEE Trans. Services Comput.*, vol. pp, no. 99, pp. 1–13, 2015.

[66] J. F. Riera, X. Hesselbach, E. Escalona, J. A. García-Espin, and E. Grasa, "On the complex scheduling formulation of virtual network functions over optical networks," in *Proc. 16th Int. Conf. Transp. Opt. Netw. (ICTON)*, Graz, Austria, Jul. 2014, pp. 1–5.

[67] J. F. Riera, E. Escalona, J. Batallé, E. Grasa, and J. A. García-Espin, "Virtual network function scheduling: Concept and challenges," in *Proc. Int. Conf. Smart Commun. Netw. Technol. (SaCoNeT)*, Vilanova i la Geltrú, Spain, Jun. 2014, pp. 1–5.

[68] R. Mijumbi *et al.*, "Design and evaluation of algorithms for mapping and scheduling of virtual network functions," in *Proc. 1st IEEE Conf. Netw. Softwarization (NetSoft)*, London, U.K., Apr. 2015, pp. 1–9.

[69] A. Fischer *et al.*, "Alevin—A framework to develop, compare, and analyze virtual network embedding algorithms," *Open Access J. Electron. Commun. EASST*, vol. 37, pp. 1–12, Mar. 2011.

[70] M. C. Luizelli, L. R. Bays, L. S. Buriol, M. P. Barcellos, and L. P. Gaspary, "Piecing together the NFV provisioning puzzle: Efficient placement and chaining of virtual network functions," in *Proc. IFIP/IEEE IM*, Ottawa, ON, Canada, May 2015, pp. 98–106.

[71] R. Cohen, L. Lewin-Eytan, J. S. Naor, and D. Raz, "Near optimal placement of virtual network functions," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr./May 2015, pp. 1346–1354.

[72] L. A. Wolsey, *Integer Programming*, vol. 42. New York, NY, USA: Wiley, 1998.

[73] A. Makhorin *et al.* (Jan. 2012). *GLPK, GNU Linear Programming Kit*. [Online]. Available: http://www.gnu.org/software/glpk

[74] *ILOG CPLEX Optimizer: High-Performance Software for Mathematical Programming and Optimization*, IBM, Armonk, NY, USA, Jun. 2012.

[75] H. Moens and F. De Turck, "VNF-P: A model for efficient placement of virtualized network functions," in *Proc. 10th Int. Conf. Netw. Service Manag. (CNSM)*, Rio de Janeiro, Brazil, Nov. 2014, pp. 418–423.

[76] A. Gupta, M. F. Habib, P. Chowdhury, M. Tornatore, and B. Mukherjee, "On service chaining using virtual network functions in network-enabled cloud systems," in *Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS)*, Kolkata, India, Dec. 2015, pp. 1–3.

[77] R. Riggio, T. Rasheed, and R. Narayanan, "Virtual network functions orchestration in enterprise WLANs," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manag. (IM)*, Ottawa, ON, Canada, May 2015, pp. 1220–1225.

[78] E.-G. Talbi, *Metaheuristics: From Design to Implementation*, vol. 74. Hoboken, NJ, USA: Wiley, 2009.

[79] R. Riggio *et al.*, "Virtual network functions orchestration in wireless networks," in *Proc. 11th Int. Conf. Netw. Service Manag. (CNSM)*, Barcelona, Spain, Nov. 2015, pp. 108–116.

[80] J. F. Riera, X. Hesselbach, M. Zotkiewicz, M. Szostak, and J.-F. Botero, "Modelling the NFV forwarding graph for an optimal network service deployment," in *Proc. IEEE ICTON*, Budapest, Hungary, Jul. 2015, pp. 1–4.

[81] M. Ghaznavi *et al.*, "Elastic virtual network function placement," in *Proc. IEEE CloudNet*, Niagara Falls, ON, Canada, Oct. 2015, pp. 255–260.

[82] R. Bruschi, A. Carrega, and F. Davoli, "A game for energy-aware allocation of virtualized network functions," *J. Elect. Comput. Eng.*, vol. 2016, Jan. 2016, Art. no. 4067186.

[83] VNREAL. (May 2015). *ALEVIN—ALgorithms for Embedding VIrtual Networks*. [Online]. Available: http://alevin.sf.net

[84] S. Clayman, E. Maini, A. Galis, A. Manzalini, and N. Mazzocca, "The dynamic placement of virtual network functions," in *Proc. IEEE NOMS*, Kraków, Poland, May 2014, pp. 1–9.

[85] M. Xia, M. Shirazipour, Y. Zhang, H. Green, and A. Takacs, "Network function placement for NFV chaining in packet/optical datacenters," *J. Lightw. Technol.*, vol. 33, no. 8, pp. 1565–1570, Apr. 15, 2015.

[86] B. Martini, F. Paganelli, P. Cappanera, S. Turchi, and P. Castoldi, "Latency-aware composition of virtual functions in 5G," in *Proc. 1st IEEE Conf. Netw. Softwarization (NetSoft)*, London, U.K., Apr. 2015, pp. 1–6.

[87] 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 *Proc. 1st IEEE Conf. Netw. Softwarization (NetSoft)*, London, U.K., Apr. 2015, pp. 1–9.

[88] A. Mohammadkhan *et al.*, "Virtual function placement and traffic steering in flexible and dynamic software defined networks," in *Proc. IEEE LANMAN*, Beijing, China, Apr. 2015, pp. 1–6.

[89] J. Soares and S. Sargento, "Optimizing the embedding of virtualized cloud network infrastructures across multiple domains," in *Proc. IEEE Int. Conf. Commun. (ICC)*, London, U.K., Jun. 2015, pp. 442–447.

[90] T. Lukovszki and S. Schmid, "Online admission control and embedding of service chains," in *Proc. SIROCCO*, Montserrat, Spain, Jul. 2015, pp. 104–118.

[91] R. Mijumbi, J. Serrat, J.-L. Gorricho, J. Rubio-Loyola, and S. Davy, "Server placement and assignment in virtualized radio access networks," in *Proc. IEEE CNSM*, Barcelona, Spain, Nov. 2015, pp. 398–401.

[92] M. Bagaa, T. Taleb, and A. Ksentini, "Service-aware network function placement for efficient traffic handling in carrier cloud," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Istanbul, Turkey, Apr. 2014, pp. 2402–2407.

[93] P. Bellavista *et al.*, "Virtual network function embedding in real cloud environments," *Comput. Netw.*, vol. 93, pp. 506–517, Dec. 2015.

[94] T. Lin, Z. Zhou, M. Tornatore, and B. Mukherjee, "Optimal network function virtualization realizing end-to-end requests," in *Proc. IEEE Glob. Commun. Conf. (IEEE GLOBECOM)*, San Diego, CA, USA, Dec. 2015, pp. 1–6.

[95] S. Sahhaf *et al.*, "Scalable architecture for service function chain orchestration," in *Proc. 4th Eur. Workshop Softw. Defined Netw. (EWSDN)*, Bilbao, Spain, Sep. 2015, pp. 19–24.

[96] B. Németh, J. Czentye, G. Vaszkun, L. Csikor, and B. Sonkoly, "Customizable real-time service graph mapping algorithm in carrier grade networks," in *Proc. IEEE Conf. Netw. Function Virtualization Softw. Defined Netw.*, San Francisco, CA, USA, Nov. 2015, pp. 28–30.

[97] S. Sahhaf *et al.*, "Network service chaining with optimized network function embedding supporting service decompositions," *Comput. Netw.*, vol. 93, pp. 492–505, Dec. 2015.

[98] M. Bouet, J. Leguay, and V. Conan, "Cost-based placement of vDPI functions in NFV infrastructures," in *Proc. IEEE NetSoft*, London, U.K., Apr. 2015, pp. 1–9.

[99] A. Baumgartner, V. S. Reddy, and T. Bauschert, "Combined virtual mobile core network function placement and topology optimization with latency bounds," in *Proc. 4th Eur. Workshop Softw. Defined Netw. (EWSDN)*, Bilbao, Spain, Sep. 2015, pp. 97–102.

[100] T. Lin, Z. Zhou, M. Tornatore, and B. Mukherjee, "Demand-aware network function placement," *J. Lightw. Technol.*, vol. 34, no. 11, pp. 2590–2600, Jun. 1, 2016.

[101] R. Riggio, A. Bradai, D. Harutyunyan, T. Rasheed, and T. Ahmed, "Scheduling wireless virtual networks functions," *IEEE Trans. Netw. Service Manag.*, vol. 13, no. 2, pp. 240–252, Jun. 2016.

[102] M. Ghaznavi, N. Shahriar, R. Ahmed, and R. Boutaba, "Service function chaining simplified," *CoRR*, vol. abs/1601.00751, 2016. [Online]. Available: http://arxiv.org/abs/1601.00751

[103] Q. Zhang, X. Wang, I. Kim, P. Palacharla, and T. Ikeuchi, "Vertex-centric computation of service function chains in multi-domain networks," in *Proc. IEEE NetSoft*, Seoul, South Korea, Jun. 2016, pp. 211–218.

[104] I. Jang, S. Choo, M. Kim, S. Pack, and M.-K. Shin, "Optimal network resource utilization in service function chaining," in *Proc. IEEE NetSoft Conf. Workshops (NetSoft)*, Seoul, South Korea, Jun. 2016, pp. 11–14.

[105] T.-W. Kuo, B.-H. Liou, K. C.-J. Lin, and M.-J. Tsai, "Deploying chains of virtual network functions: On the relation between link and server usage," in *Proc. IEEE INFOCOM 35th Annu. IEEE Int. Conf. Comput. Commun.*, San Francisco, CA, USA, Apr. 2016, pp. 1–9.

[106] R. V. Rosa, C. E. Rothenberg, and R. Szabo, "VBaaS: VNF benchmark-as-a-service," in *Proc. 4th Eur. Workshop Softw. Defined Netw.*, Bilbao, Spain, Sep./Oct. 2015, pp. 79–84.

[107] W.-P. Lu, "Validation as a service (VAAS)," in *Proc. IEEE SDN EIT Digit. Federated Testbeds Workshop*, San Francisco, CA, USA, May 2016, pp. 1–18.

[108] R. Mijumbi, J. Serrat, J.-L. Gorricho, and J. Rubio-Loyola, "On the energy efficiency prospects of network function virtualization," *CoRR*, vol. abs/1512.00215, 2015.

Juliver Gil Herrera received the M.Sc. degree in computer engineering from the Polytechnic University of Turin, Italy, in 2013. He is currently pursuing the Ph.D. degree in electronic and computer engineering with the Electronics and Telecommunications Engineering Department, University of Antioquia, Colombia. His main research and professional interests involve network optimization, NFV, SDN, and network virtualization. He also is interested in future Internet architecture and IoT.

Juan Felipe Botero received the Ph.D. degree in telematics engineering from the Technical University of Catalonia, Spain, in 2013. He is an Assistant Professor with the Electronics and Telecommunications Engineering Department, University of Antioquia, Colombia. In 2013, he joined GITA (a research group on applied telecommunications) with the Electronics and Telecommunications Engineering Department. His main research interests include quality of service, software defined networking, network virtualization, and resource allocation.