A Marketplace as a Scalable Solution to the Orchestration Problem in SDN/NFV Networks

Salvatore D'Oro, Laura Galluccio, *Member, IEEE*, Sergio Palazzo, *Senior Member, IEEE*, Giovanni Schembra

***Abstract*—In the SDN/NFV ecosystem, network services are provided as single Virtual Network Functions (VNFs) or chains of them, each instantiated and executed on dedicated servers. So far, the chaining of those virtual functions, which is also known as the *service chain composition problem*, has been mostly performed by Telco Operators (TOs) for the great advantages they receive in terms of Capex and OpEX. However, such a fully centralized approach generally results in solutions which do not scale well with the number of customers. The aim of this paper is to provide a distributed, scalable and efficient solution to the service chaining problem. Specifically, we develop an VNF marketplace system where third-party VNF providers sell VNF as a service (VNFaaS) and adapt their pricing policies according to network dynamics. Also, we leverage on game theory to provide a (theoretically proven) efficient distributed solution which accounts for monetary costs, communication latencies and congestion of computational resources.**

***Index Terms*—Softwarized Networks, Orchestration, Resource Management, Marketplace Model, Game Theory.**

I. INTRODUCTION

In the last decade, network softwarization has become the focus of networking research and development worldwide. To this purpose, key elements today are represented by two network paradigms, that is, Software Defined Networking (SDN) and Network Functions Virtualization (NFV).

The main goal of SDN [1–3] is to move the decision making process of the network from network nodes to the SDN Controller, in order to separate the data plane from the control plane.

On the other hand, the NFV paradigm [4–6] has emerged in the last few years by leveraging on the concepts of cloud and virtualization technologies to enable cost-efficient and flexible network softwarization. The main concept behind the joint SDN/NFV paradigm is virtualizing network functions by hosting them on virtual machines (VM) launched on general-purpose physical servers inside a Network Function Virtualized Infrastructure (NFVI) [7, 8], and using SDN to steer traffic flows through network nodes where the instances of VNFs required by them are running.

Key elements for the design of these systems are allocation, management and orchestration of network resources, that result more challenging as compared to scenarios of legacy networks. In this context, the choice of how many instances using for each VNF, how many servers, in the following referred to as *VNF Servers*, maintaining active in the network, which servers using to host the instances of each VNF, the amount of hardware resources dedicating in each server in terms of computing, storage and networking, are strategic choices in the lifecycle of softwarized networks based on the SDN/NFV paradigm. An additional problem, referred to as the *Service Function Chaining problem* (SFCP) [9], is how chaining VNF instances to realize more complex network services (NSs).

As recognized in the existing literature [10, 11], this is an NP-hard problem. To this purpose, some sub-optimal centralized solutions in the network Orchestrator have been proposed in the previous literature [12–15]. Nevertheless, those approaches mainly rely on the non-realistic assumption that full-knowledge is available at the network Orchestrator.

In opposition to the VNF market, where VNF provision and network orchestration are centralized by the TO, the VNF marketplace architecture has been proposed in [16]. According to the marketplace definition, third-party VNF providers, i.e., *VNF Servers*, can use their hardware and software resources to sell VNFs as a service (VNFaaS), thus helping the TO in providing network services in an efficient and scalable way.

The aim of this paper is to provide a distributed, scalable, and efficient solution to the SFCP. Accordingly, we first propose a practical implementation of the VNF marketplace envisioned in [16]. Then, to achieve scalability, we leverage on game theory to design a low-complexity user-centric distributed chaining algorithm. The proposed algorithm allows network customers to build their own service chains based on their individual requirements, such as monetary costs, communication latency and congestion of each VNF Server. The proposed approach is shown to provably converge towards an effient solution of the SFCP. In addiction, we design a pricing mechanism such that VNF Servers can autonomously decide the price to be applied to each VNFs.

The rest of this paper is organized as follows. In Section II we describe the reference system. In Section III the Marketplace model is discussed. In Section IV the numerical results are shown. Finally, in Section V some conclusions are drawn.

II. REFERENCE SYSTEM

The scenario considered as reference in this paper is sketched in Fig. 1. It consists of a TO network that provides customers with NSs according to the SDN/NFV paradigm. A NS can be either composed of a single VNF, or by a chain of them. The main roles in the system are played by the Users, the VNF Servers and the Orchestrator.

All authors are with the Dipartimento di Ingegneria Elettrica, Elettronica ed Informatica, University of Catania, Italy;

E-Mail: {name.surname}@dieei.unict.it

This work has been partially supported by the H2020 INPUT project.

This paper has been accepted for publication on the *3rd IEEE Conference on Network Softwarization (IEEE NetSoft 2017)*. This is a preprint version of the accepted paper. The final paper will be available soon.

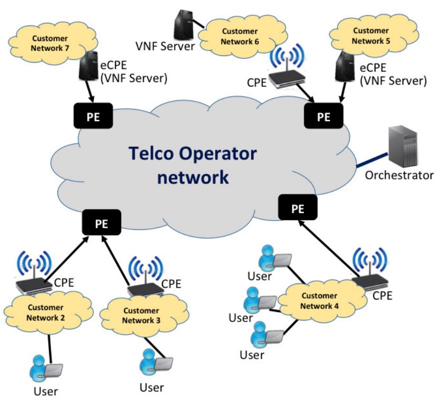


Fig. 1. The reference network scenario.

Users generate *traffic flows*. Network services with the associated levels of quality can be requested by single users and for each of their flows. Alternatively, when many users enter the network through an access network in a business environment, e.g. a University Campus, or owned by a third party provider, e.g., a 3G/4G mobile access network, NSs provided to network customers are usually predefined by the access network provider. In some cases, traffic can be divided in classes, and a network service with the associated QoS level is assigned to each traffic class.

VNF Servers are NFV-compliant nodes [7, 8] owned by network customers that have decided to participate to the NFV marketplace by providing VNFaaS in order to obtain some economic benefits by serving the TO network. A VNF Server can be either a stand-alone computer, a set of servers organized as a data center, whose resources are partially or totally dedicated to run VNFs, or an enhanced Customer Premise Equipment (eCPE) node [17], that is, a CPE device (for example, a residential home gateway) that is able to run VNFs in a virtualized environment. A VNF Server can provide more than one VNF and decide the selling prices autonomously.

A very important role in an SDN/NFV system is played by the *Network Orchestrator* that, running on a dedicated server, is responsible of management and orchestration of the whole system. According to the Management and Orchestration (MANO) specifications defined for NFV networks in [8], in a legacy SDN/NFV system its main target is to instantiate VMs to run VNFs on the available VNF Servers, decide the amount of resources that the VNF Servers have to reserve to each VNF in terms of computing, storage and network resources, and arranging the NFVI in such a way to provide customers with the required network services with the relative expected quality. In addition, for network services composed by a single VNF, the Network Orchestrator decides how many VNF instances have to be executed in the network, and the VNF Servers that have to run them. Likewise, in the case of more complex network services that are constituted as chains of different VNFs, the Network Orchestrator is in charge of the Service Chain Composition task, which consists in chaining the running VNF instances to realize instances of NSs with different levels of QoS.

The above tasks constitute together a very hard optimization problem with many constraints. The target of this paper, as said so far, is to approach this problem by introducing the marketplace model, so distributing this task, which is centralized in legacy SDN/NFV networks, to the network customers. In fact, decisions regarding resource allocation to run VNFs are no longer in charge of the Network Orchestrator because are locally decided by the VNF Servers that behave as *Sellers* in the marketplace. On the other hand, creation of chains of VNF instances is not in charge of the Orchestrator since each User will choose the VNF instances to be used, and chain them to create the network service that satisfies its requirements in terms of both QoS and price.

In order to relieve Users to directly compose by themselves the VNF chains, we introduce a new entity, the *Network Service Broker* (NSB). An NSB is a software application that represents flows generated by Users belonging to a delimited portion of network, which in the following will be indicated as NSB Scope. An *NSB Scope* can be either 1) a user terminal, 2) a local area network (e.g. the home LAN in a residential environment, or a switched LAN in a business environment), or 3) an access network of third party providers, for example the radio access network (RAN) of 3G/4G mobile networks. All the NSBs in the network interact with each other as *players* of a game that will be defined in the sequel.

In the following, without losing in generality and to simplify readability, we will refer to a *traffic flow* as either a single traffic flow, or an aggregate of traffic flows that, although generated by different Users or different applications of the same User, have the same ingress and egress edge nodes, and require the same NS with the same level of quality.

III. THE MARKETPLACE MODEL

Let us now discuss the interactions between VNF Servers and NSBs. To simplify the notation, in the following we will focus on one NS, indicated as $\mathcal{Y}$ [1]. We will indicate the set of NSBs devoted to $\mathcal{Y}$ as $\mathcal{U}$.

Decisions taken by VNF Servers and NSBs depend on both individualistic interests, e.g. maximize (minimize) their own utility (cost), and decisions taken by counterparts, e.g. opponents' strategies.

In the following, we show how the above interactions among VNF Servers and NSBs can be modeled by exploiting tools from both game and approximation theories. Specifically, in Section III-A we first describe the considered model, and summarize the relevant assumptions. Then, in Section III-B we show that the SFCP can be formulated as a congestion game [18] played by the NSBs. Finally, to model the interactions among the VNF Servers, a simple, but effective, pricing mechanism based on stochastic approximation theory is presented in Section III-C.

---

[1] Note that the more general case where many network services are provided to network users can be tackled with the same approach presented in this paper by exploiting the multi-tenancy property provided by network slicing. This allows to virtually create independent network and computing slices for each service chain without mixing traffic generated by NSB threads requesting different chains.

A. Model Notation and Assumptions

Let us define the considered NS as an ordered sequence of F VNFs, i.e. $\mathcal{Y} = \{1, 2, \dots, F\}$. Let $N = |\mathcal{U}|$ be the number of the NSBs in the network whose flows request the network service $\mathcal{Y}$. For each VNF $f \in \mathcal{Y}$, the entities that will participate in the marketplace for it are: 1) the VNF Servers that run f, as sellers; 2) the NSBs that need f for their flows, as representative of buyers; 3) the TO that has decided to provide its customers with f and that wants to gain some economic benefit from it.

We assume that VNFs are executed on a set $\mathcal{S} = \{1, 2, \dots, V\}$ of V active VNF Servers. Furthermore, to execute an instance of the VNF f on its server, we assume that v fixes a *VNF price* $p_{v,f}$ for each VNF $f \in \mathcal{Y}$.

Let $\lambda_i > 0$ be the bit rate of the flow handled by the i-th NSB, and let s_i and t_i be the ingress and the egress nodes of its handled flow, respectively. Hence, the position in the network of each NSB is determined by the pair (s_i, t_i).

To consider the communication latencies between NSBs and VNF Servers, we define the following latency matrices. Let $\mathbf{D} = (d_{v',v''})_{v',v''\in\mathcal{S}}$ be the *Server-to-Server latency matrix* containing the latencies between all VNF Servers in the network. Similarly, let $\mathbf{D}^{\text{in}} = (d^{\text{in}}_{s_i,v})_{i\in\mathcal{U},v\in\mathcal{S}}$ be the *ingress-to-Server latency matrix* which contains the latencies between all ingress nodes and VNF Servers. Finally, let $\mathbf{D}^{\text{out}} = (d^{\text{out}}_{v,t_i})_{v\in\mathcal{S},i\in\mathcal{U}}$ be the *Server-to-egress latency matrix* containing all latencies between VNF Servers and egress nodes.

To simplify the notation, in the following we use $d_{a,b}$ to identify the latency between nodes a and b, where a, b can be either VNF Servers in $\mathcal{S}$, ingress or egress nodes. Also, we assume that all VNF Servers are interconnected to each other.

Let $\mathcal{W}$ be the set of all possible configurations for the NS $\mathcal{Y}$. Thus, the cardinality of $\mathcal{W}$ is $|\mathcal{W}| = V^F$. Furthermore, let $\mathbf{w}_i \in \mathcal{W}$ be the NS configuration chosen by the NSB representing the flow $i \in \mathcal{U}$. Specifically, $\mathbf{w}_i$ is defined as the F-tuple $\mathbf{w}_i = (w_i(1), w_i(2), \dots, w_i(F))$, where $w_i(f) \in \mathcal{S}$ is the VNF Server which has been chosen by the NSB i to execute the function $f \in \mathcal{Y}$.

For a given NS configuration $\mathbf{w}_i$ chosen by NSB i, the corresponding price $c_i^{(\mathcal{P})}(\mathbf{w}_i)$ and overall latency $c_i^{(\mathcal{T})}(\mathbf{w}_i)$ are defined as

$$c_i^{(\mathcal{P})}(\mathbf{w}_i) = \sum_{f=1}^{F} p_{w_i(f),f} \tag{1}$$

$$c_i^{(\mathcal{T})}(\mathbf{w}_i) = d_{i,w_i(1)} + \sum_{f=2}^{F} d_{w_i(f-1),w_i(f)} + d_{w_i(F),i} \tag{2}$$

where the term $d_{s_i,w_i(1)} \in \mathbf{D}^{\text{in}}$ is the latency between the ingress node s_i and the server $w_i(1) \in \mathcal{S}$, the term $d_{w_i(f-1),w_i(f)} \in \mathbf{D}$ is the inter-server latency between servers $w_i(f-1)$ and $w_i(f)$, and the term $d_{w_i(F),t_i} \in \mathbf{D}^{\text{out}}$ is the latency between the server $w_i(F)$ and the egress node t_i.

Since different flows can use the same instance of the VNF f, they will experience a congestion level due to the load on the server where f is running. Let $\mathbf{w}_{-i} = (\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_{i-1}, \mathbf{w}_{i+1}, \dots, \mathbf{w}_N)$, and let $\Gamma^f(\mathbf{w}_i, \mathbf{w}_{-i}, v)$ be the set of NSBs in $\mathcal{U}$ which have chosen the server $v \in \mathcal{S}$ to receive the function $f \in \mathcal{Y}$. Let us indicate the NS configuration chosen by the NSB j as $\mathbf{w}_j$, and the set of the NS configurations chosen for all the NSBs requiring the considered network service $\mathcal{Y}$ as $(\mathbf{w}_i, \mathbf{w}_{-i}) \in \mathcal{W}^N$, where $\mathcal{W}^N$ stands for the N-ary Cartesian power of the set $\mathcal{W}$.

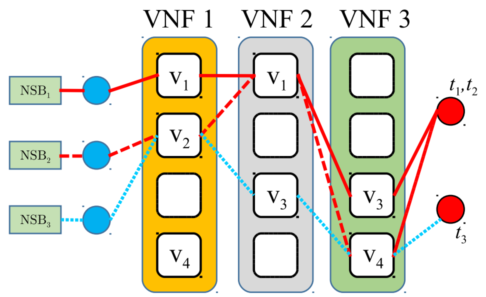


Fig. 2. An illustrative example of a possible service chain configuration.

The congestion level experienced by the flow corresponding to the NSB i for function f on the chosen server $w_i(f)$ is

$$\delta_f(\mathbf{w}_i, \mathbf{w}_{-i}) = \sum_{j\in\Gamma^f(\mathbf{w}_i,\mathbf{w}_{-i},w_i(f))} \lambda_j \tag{3}$$

where $\delta_f(\mathbf{w}_i, \mathbf{w}_{-i}) = \delta_f(\mathbf{w}_j, \mathbf{w}_{-j})$ if $w_j(f) = w_i(f)$.

By exploiting (3), we have that the overall congestion level experienced by the flow of NSB $i \in \mathcal{U}$ is $c_i^{(\mathcal{C})}(\mathbf{w}_i, \mathbf{w}_{-i}) = \sum_{f=1}^{F} \delta_f(\mathbf{w}_i, \mathbf{w}_{-i})$.

Accordingly, the total cost experienced by the flow represented by the NSB $i \in \mathcal{U}$ can be expressed by the following *cost function*:

$$\mathcal{C}_i(\mathbf{w}_i, \mathbf{w}_{-i}) = c^{(\mathcal{C})}(\mathbf{w}_i, \mathbf{w}_{-i}) + \gamma_i \left(c_i^{(\mathcal{T})}(\mathbf{w}_i) + \beta_i c_i^{(\mathcal{P})}(\mathbf{w}_i) \right) \tag{4}$$

where γ_i and β_i are two non-negative NSB-specific weighs.

For the sake of illustration, the considered service chain model is shown in Fig. 2. Specifically, we present an illustrative system configuration where $F = 3$ VNFs compose the NS $\mathcal{Y} = \{1, 2, 3\}$, $V = 4$ VNF Servers are in the network, and $N = 3$ NSBs choose the NS configuration for their flows. As shown in the figure, some servers may not provide functions in the chain, and it may happen that some NSB in $\mathcal{U}$ share the same ingress and/or egress nodes. In the example presented in the figure, we have service chains chosen for the three flows, $\mathbf{w}_1$, $\mathbf{w}_2$ and $\mathbf{w}_3$, represented by solid, dotted and dashed lines, respectively. Flows 2 and 3 share the same server v_2 to execute the VNF 1, and all of the three flows share server v_3 to execute the VNF 3. Therefore, from (3) we have that the server load at v_2 w.r.t. VNF 1 is equal to $\lambda_2 + \lambda_3$. Similarly, the server load on v_3 w.r.t. VNF 3 is $\lambda_1 + \lambda_2 + \lambda_3$, while the server load at v_1 w.r.t. VNF 1 is equal to λ_1, as only flow 1 has selected v_1 to execute VNF 1.

B. Game $\mathcal{G}$ modeling the interactions among NSBs

Each NSB aims to create service chains for their flows, while selfishly minimizing the cost function in (4). Furthermore, NSBs have conflicting interests as they compete with each other to use the same resources, and are not expected to cooperate. Thus, it has been shown that their interactions can be modeled as the weighted congestion game $\mathcal{G} =$

$\big(\mathcal{U}, (\lambda_i)_{i\in\mathcal{U}}, \mathcal{S}_f, \mathcal{W}^N, (\mathcal{C}_i)_{i\in\mathcal{U}}\big)$ [19], where $\mathcal{U}$ is the set of *players*, the *weights* of the congestion game are the bit rates λ_i of the relative flows, the set $\mathcal{S}_f$ of servers which provide VNF $f \in \mathcal{Y}$ corresponds to the set of *resources* that can be selected by players, $\mathcal{W}$ are all the possible NS configurations that can be chosen by the player $i \in \mathcal{U}$, and $\mathcal{C}_i$ is the utility function of the i-th player. In the following of this paper, we will refer to the VNF Servers as the resources of the congestion games, and we use terms NSB and player interchangeably.

It can be shown that game $\mathcal{G}$ admits at least one Nash Equilibrium (NE), which is a strategy profile $(\mathbf{w}_1^*, \mathbf{w}_2^*, \dots, \mathbf{w}_N^*) \in \mathcal{W}^N$ where no player has incentive to deviate unilaterally. Furthermore, it is possible to prove that the performance of the system at a NE are close to optimal, and any NE can be computed in polynomial time through iterative distributed algorithms. Due to the lack of space, proofs are omitted in this paper. However, we refer the interested reader to [19] for a more detailed report.

C. Pricing model for VNF Servers

For each traffic flow traversing a given VNF Server, this server has to allocate a given amount of computing and storage resources, and this represents a cost for the VNF Server. Considering the generic VNF Server $v \in \mathcal{S}$, we will refer to the incremental cost incurred by the VNF Server v to guarantee the required resources to a new flow requiring function f as $\rho_{v,f}$. For example, the cost $\rho_{v,f}$ may be an energy cost as in [17] that depends on the price applied by the energy provider.

Therefore, the cost for a VNF Server v to support all the traffic flows managed by the NSBs in $\mathcal{U}$, $C_{v,f}$, is proportional to the number of NSBs $n_{v,f}$ that choose it to compose a service chain for their flows, that is

$$C_{v,f} = \rho_{v,f} \cdot n_{v,f} \tag{5}$$

On the other hand, the revenue for the VNF Server v associated to the provision of VNF f is proportional to both the number of NSBs $n_{v,f}$ that choose it to compose a service chain for their flows, and the price $p_{v,f}$ applied by this VNF Server. The revenue of the VNF Server v related to the provision of VNF f is thus

$$R_{v,f} = p_{v,f} \cdot n_{v,f} \tag{6}$$

The profit achieved by the VNF Server v w.r.t. VNF f can be written as

$$\Pi_{v,f}^{(\mathcal{S})}(\mathbf{p}_v) = R_{v,f} - \alpha C_{v,f} = n_{v,f}\,(p_{v,f} - \alpha\rho_{v,f}) \tag{7}$$

where α is a weight that trade-offs revenues and costs.

Accordingly, the *profit function* $\Pi_v^{(\mathcal{S})}(\mathbf{p}_v)$ for a given VNF Server v is defined as $\Pi_v^{(\mathcal{S})}(\mathbf{p}_v) = \sum_{f\in\mathcal{Y}} \Pi_{v,f}^{(\mathcal{S})}(\mathbf{p}_v) = \sum_{f\in\mathcal{Y}} n_{v,f}\,(p_{v,f} - \alpha\rho_{v,f})$, where $\mathbf{p}_v = (p_{v,f})_{f\in\mathcal{Y}}$ is the *price vector* for VNF Server v.

We assume that all VNF Servers are rational, i.e., they aim at achieving positive profit. Accordingly, in the following we assume that the VNF price $p_{v,f}$ is chosen such that $p_{v,f} > \alpha\rho_{v,f}$.

$\Pi_v^{(\mathcal{S})}(\mathbf{p}_v)$ depends on the number of NSBs $n_{v,f}$, which is the result of the interactions among NSBs described in Section III-B. Specifically, for any given *price profile* $\mathbf{p} = (\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_V)$, let $\mathbf{w}^*$ be the NE of $\mathcal{G}$. The number of NSBs $n_{v,f}$ that at the NE $\mathbf{w}^*$ has chosen the VNF Server v to receive function f is defined as

$$n_{v,f} = |\Gamma^f(\mathbf{w}^*, v)| \tag{8}$$

It is worth noting that, in general, lower values of the VNF price $p_{v,f}$ do not necessarily attract more NSBs and produce more profit. In fact, when the price is low, a higher number of NSBs are expected to select the same VNF Server, which also produces high congestion levels thus pushing NSBs in avoiding congested VNF Servers by selecting other servers with possibly higher prices. Accordingly, the selection of the VNF price $p_{v,f}$ is not an easy task as it depends on the non-cooperative dynamics of the game $\mathcal{G}$.

To account for those dynamics, in the following we assume that the price $p_{v,f}$ is periodically updated every Δ seconds. For each VNF Server v and VNF f, we assume that the VNF price $p_{v,f}$ is updated according to the following stochastic approximation mechanism:

$$p_{v,f}(t+1) = p_{v,f}(t) + \sigma_v(t)\,[n_{v,f}(t) - n_{v,f}(t-1)] \tag{9}$$

where t is the t-th iteration of the mechanism, and $\sigma_v(t)$ is the step-size of the stochastic procedure which here is assumed to be defined as $\sigma_v(t) = 1/t$ for all $v \in \mathcal{S}$.

From (9), if the number of NSBs connected to a VNF Server has been increased, i.e., $n_{v,f}(t) > n_{v,f}(t-1)$, then the price at the next iteration is increased as well, i.e., $p_{v,f}(t+1) > p_{v,f}(t)$. Otherwise, if $n_{v,f}(t) < n_{v,f}(t-1)$, we have that $p_{v,f}(t+1) < p_{v,f}(t)$.

Furthermore, to achieve positive profit, the condition $p_{v,f}(t+1) > \alpha\rho_{v,f}$ has to be satisfied. It is worth noting that the stochastic procedure in (9) might produce a VNF price $p_{v,f}(t+1)$ that violates the latter constraint, which does not satisfy the rationality assumption. Accordingly, to ensure a positive profit to each VNF Server, a minimum value $p_{v,f}(t+1) = \alpha\rho_{v,f} + \epsilon$ is considered at each iteration of (9).

IV. NUMERICAL RESULTS

In our simulations, we have assumed that all flows managed by the NSBs are uniform, that is, for each NSB $i \in \mathcal{U}$, we consider a transmission data rate $\lambda_i = \lambda = 10$ Mbit/s. Furthermore, and unless stated otherwise, we assume that the service chain is composed of $F = 5$ VNFs, and we set $\beta_i = \beta = 5$ for all NSBs. The weighing parameter α in (7) is set such that each VNF Server equally weighs revenues and costs, i.e., $\alpha = 1$. Also, we assume that the cost $\rho_{v,f}$ in (5) is $\rho_{v,f} = 10$ price units for all $v \in \mathcal{S}$ and $f \in \mathcal{Y}_v$.

We assume that the VNF price $p_{v,f}$ is uniformly distributed in the interval $[1, 500]$, $\gamma = 1$ and the step-size in (9) is $\sigma_v(t) = 100/t$ for all $v \in \mathcal{S}$. The elements of the latency matrices $\mathbf{D}$, $\mathbf{D}^{\text{in}}$ and $\mathbf{D}^{\text{out}}$ are generated according to a Gamma distribution with mean value of 8 ms and variance equal to 0.004 ms [19]. The VNF marketplace and the chaining algorithm have been implemented and tested in MATLAB.

At the beginning of the game, we assume that the initial pricing profile chosen by each VNF Server is as shown in the first subplot in Fig. 3. In the simulated scenario, we have

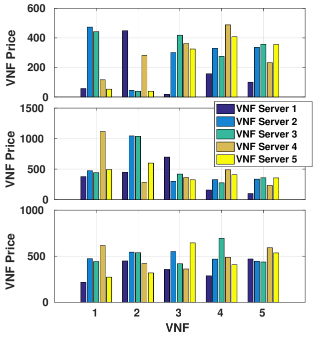
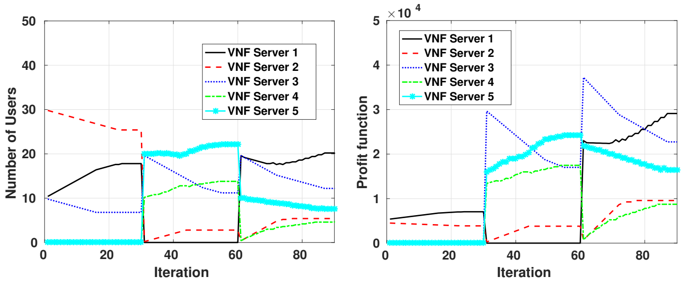


Fig. 3. Dynamic evolution of the VNF prices $p_{v,f}$, one subplot for each period at constant price.



(a)

(b)

Fig. 4. a) Dynamic evolution of the average number of users attached to each VNF Server; b) Dynamic evolution of the profit $\Pi_v^{(S)}(\mathbf{p}_v)$ of each VNF Server.

assumed that NSBs exploit the distributed service chaining algorithm proposed in [19]. Thus, at each iteration of the algorithm, the NSBs adapt their strategy by selecting the most profitable VNF Servers for their handled flows. In Fig. 4(a), we show the average number of NSBs that select each VNF Server to run the considered VNFs. Fig. 4(a) shows that the number of users converges towards a NE of the game until the pricing policy is updated again. Specifically, the pricing policies are updated at iterations 30 and 60, and are shown in the second and third plots of Fig. 3, respectively.

Instead, in Fig. 4(b) we show how the profit $\Pi_v^{(S)}(\mathbf{p}_v)$ of each VNF Server $v \in \mathcal{S}$ varies at each iteration according to the number $\sum_{f \in \mathcal{Y}_v} n_{v,f}$ of NSBs that are connected to it. As expected, the profit is proportional to the number of users.

V. CONCLUSIONS AND FUTURE WORK

The centralized nature of the current VNF market architecture does not completely exploit the flexible and effective management of network resources envisioned for the SDN/NFV paradigm. Such an inefficient use of resources can be partially tackled by exploiting the VNF marketplace architecture which allows third-party VNF providers to participate in the VNF provisioning market. In this paper, we have shown that scalable and efficient solutions to the SFCP can be designed by effectively exploiting the VNF marketplace architecture. In addiction, we have shown that game-theory can be proficiently exploited to develop user-centric distributed algorithms.

The proposed solution allows network customers to individually build their own service chain configuration according to their requirements in terms of communication latencies, congestion on the network resources and the monetary costs to access the marketplace and build the chain. Also, we have shown that dynamic pricing of the VNFs is feasible. Specifically, to maximize the expected profit, dynamic pricing policies can be effectively exploited by the third-party VNF Servers by adapting their pricing strategies to network changes and customers' time-varying behavior.

As future work, the interactions between the VNF Servers and NSBs will be investigated in more detail. For example, dynamic pricing policies will be investigated to derive optimal strategies to improve customers' performance and VNF Servers' profit. Also, a software implementation of the proposed distributed architecture will be provided.

REFERENCES

[1] Open Networking Foundation, "Software-defined networking: The new norm for networks," *ONF White Paper*, 2012.
[2] ONF, "Software-defined networking: The new norm for networks." *White Paper 2012.*
[3] C. Monsanto, J. Reich, N. Foster, J. Rexford, and D. Walker, "Composing software defined networks." in *Proc. of NSDI*, 2013, pp. 1–13.
[4] (http://portal.etsi.org/NFV/NFV_White_Paper.pdf) White paper on network functions virtualization.
[5] ONF, "Network functions virtualisation update." *White Paper 2 https://portal.etsi.org/nfv/nfv_white_paper2.pdf.*
[6] ——, "Network functions virtualisation update." *White Paper 3 https://portal.etsi.org/Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf.*
[7] ETSI NVF GS, Network Functions Virtualization (NFV) Infrastructure Overview, NFV-INF 001 V1.1.1, Jan 2015.
[8] ETSI NFV GS, Network Function Virtualization (NFV) Management and Orchestration, NFV-MAN 001 v0.8.1, Nov 2014.
[9] T. Nadeau and P. Quinn, "Problem Statement for Service Function Chaining," RFC 7498, Nov. 2015.
[10] T. Lukovszki, M. Rost, and S. Schmid, "It?s a match! near-optimal and incremental middlebox deployment," *ACM SIGCOMM Computer Communication Review, Vol. 46 Issue 1, January 2016.*
[11] A. Lombardo, A. Manzalini, V. Riccobene, and G. Schembra, "An analytical tool for performance evaluation of software defined networking services," *Proc. of IEEE SDNMO, Krakow, Poland*, 2014.
[12] B. Batouche, Y. Naudet, and F. Guinand, "Semantic web services composition optimized by multi-objective evolutionary algorithms," in *Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on*, May 2010, pp. 180–185.
[13] G. Canfora, M. Di Penta, R. Esposito, and M. L. Villani, "An approach for QoS-aware service composition based on genetic algorithms," in *Proceedings of the 7th annual conference on Genetic and evolutionary computation*. ACM, 2005, pp. 1069–1075.
[14] S. Sahhaf, W. Tavernier, M. Rost, S. Schmid, D. Colle, M. Pickavet, and P. Demeester, "Network service chaining with optimized network function embedding supporting service decompositions," *Computer Networks*, vol. 93, pp. 492–505, 2015.
[15] P. Wang, J. Lan, X. Zhang, Y. Hu, and S. Chen, "Dynamic function composition for network service chain: Model and optimization," *Computer Networks*, vol. 92, pp. 408–418, 2015.
[16] G. Xilouris, E. Trouva, F. Lobillo, J. Soares, J. Carapinha, M. J. McGrath, G. Gardikis, P. Paglierani, E. Pallis, L. Zuccaro *et al.*, "T-nova: A marketplace for virtualized network functions," in *Networks and Communications (EuCNC), 2014 European Conference on*. IEEE, 2014, pp. 1–5.
[17] G. Faraci and G. Schembra, "An analytical model to design and manage a green SDN/NFV CPE node," *Network and Service Management, IEEE Transactions on*, vol. 12, no. 3, pp. 435–450, Sept 2015.
[18] R. W. Rosenthal, "A class of games possessing pure-strategy nash equilibria," *International Journal of Game Theory*, vol. 2, no. 1, pp. 65–67, 1973.
[19] S. D'Oro, L. Galluccio, S. Palazzo, and G. Schembra, "Exploiting congestion games to achieve distributed service chaining in NFV networks." *IEEE Journal on Selected Areas in Communications, Vol. 35, no. 2, pp. 407 - 420, February 2017.*