SCHE2MA: Scalable, Energy-Aware, Multi-Domain Orchestration for Beyond-5G URLLC services

Anestis Dalgkitsis, Luis A. Garrido, Farhad Rezazadeh, Hatim Chergui,
Kostas Ramantas, John S. Vardakas, Christos Verikoukis

Abstract—**The evolution of Software-Defined Networking (SDN) and Network Function Virtualization (NFV) in the telecommunications industry have intensified the issues of network management at large scales. Dynamic service orchestration and adaptive resource allocation became a necessity for network operators to manage the rapid growth of users and data-intensive applications. The impact of network automation on energy consumption and overall operating costs is often overlooked. Guaranteeing strict performance constraints of Ultra-Reliable Low Latency Communication (URLLC) services while enhancing energy efficiency is one of the major critical problems of future communication networks, given the urgency to reduce carbon emissions and energy consumption. In this work, we study the problem of zero-touch Service Function Chain (SFC) orchestration for multi-domain networks, targeting the latency reduction of URLLC services while improving energy efficiency for beyond-5G networks. Specifically, we propose SCHE2MA, a Service CHain Energy-Efficient MAnagement framework based on distributed Reinforcement Learning (RL), that can intelligently deploy SFCs with shared VNFs per se into a multi-domain network. Finally, we evaluate SCHE2MA through model validation and simulation while demonstrating its ability to jointly reduce average service latency by 103.4% and energy consumption by 17.1% compared to a centralized RL solution.**

Index Terms—**Zero-touch Orchestration, Network Function Virtualization, Service Function Chain Placement, Distributed Reinforcement Learning, Energy Efficiency.**

I. Introduction

THE rapid expansion of 5G networks introduced the need for automated scaling to cope with the anticipated demand of the new high-performing communication networks. The networks grow at a rate that creates an urgent pressure for Infrastructure Providers (InPs) to adopt technologies for intelligent automated network management and orchestration. The lack of intelligent automation will cause additional costs and make it unfeasible for operators to guarantee the performance required by new 5G and Beyond-5G (B5G) network services.

The introduction of technologies such as Software-Defined Networking (SDN) and Network Function Virtualization (NFV) enabled the deployment of multiple virtual networks over a physical network infrastructure, with distinct performance and service agreement constraints. Moreover, this architecture enables the decoupling of service providers from the physical InPs. The virtualization of the network also introduced more flexibility and more possibilities to optimize energy efficiency. Mobile Edge Computing (MEC) enabled near real-time low latency but introduced additional computational resources in multiple computing domains, spread across diverse geographic locations. The deployment of MEC servers has allowed for partial or complete migration of the network services at the edge of the network [1]. Having a considerable number of MEC servers introduces additional issues of increased costs related to installation, operation, maintenance and energy consumption. Flexible service provisioning has the potential to significantly reduce not only the Capital Expenditures (CAPEX) and the Operational Expenditures (OPEX), but also the total energy consumption of the network.

The presence of virtual networks in B5G, enabled by SDN and NFV, has sparked the introduction of a new variety of services, making it possible to deploy multiple network services within a virtual network. In this context, a typical network service is built using a series of interconnected Virtual Machines (VMs) or Containers that perform a specific function, called Virtual Network Functions (VNFs). The VNFs are usually chained together, forming complex structures called Service Function Chains (SFCs) with data flows among them. The physical network of the InPs is spread in a large geographical area comprised of multiple domains across the physical network. This forces the SFCs to be deployed over massive distances with their VNFs located in distant data centers.

The deployment of SFCs is rather complex, as VNFs must be placed in an optimal setting to ensure maximum performance and minimum energy consumption while performing the service requests. However, the underlying physical network can have multiple geographically distributed domains, which affects services that have a spatial distribution that extends large distances. It is safe to conclude that the placement of service VNFs in the network holds a significant role in the performance, the quality of the offered service, the energy consumption and finally, the cost of operation and maintenance.

As the size of cellular networks is expanding and the demand for computing resources increases, the necessity for automated service management and orchestration becomes essential for InPs to offer superior services to the users. The autonomous placement of SFCs is a fundamental aspect of the Zero-touch network and Service Management (ZSM) in B5G networking. The European Telecommunications Standards Institute (ETSI) has generated guidelines regarding ZSM

Iquadrat Informatica, Carrer del Dr. Rizal 10, 08006 Barcelona, Spain (e-mail: a.dalgkitsis@iquadrat.com; l.garrido@iquadrat.com, kramantas@iquadrat.com, jvardakas@iquadrat.com).

University of Patras, University Campus, 265 04, Rion, Achaia, Greece (e-mail: cveri@ceid.upatras.gr)

Centre Tecnologic de Telecomunicacions de Catalunya (CTTC/CERCA), Av. Carl Friedrich Gauss 7, 08860 Castelldefels, Barcelona, Spain (e-mail: hchergui@cttc.es; frezazadeh@cttc.es).

and self-management on large-scale networks [2]. Researches from both academia and industry are proposing and developing state-of-the-art solutions following the guidelines by ETSI [3].

Performance optimization with ZSM orchestration is a topic well studied in modern literature. However, performance scaling is often overlooked, especially for complex trade-offs such as latency and energy minimization. Single-agent centralized solutions, not only exhibit a centralized point of failure but also bottleneck automation scaling. Large 5G networks are divided into geographically distributed domains to satisfy various remote regions. This natural separation can be utilized to create a superior distributed and decentralized SFC management and orchestration system to enable scalability. By dividing the problem into multiple local regions that cooperate, it avoids affecting the performance of a wide variety of Key Performance Indicators (KPIs) as various aspects of the network grow over time.

In a previous work [4], we have studied the problem of VNF orchestration for VNFs in single-domain networks and proposed a Deep Deterministic Policy Gradient Reinforcement Learning (RL) solution for End-to-End (E2E) service latency minimization. Afterwards, we have proposed a distributed RL-based orchestration framework capable of orchestrating multiple SFCs in multi-domain networks and optimizing the placement for Ultra-Reliable Low Latency Communications (URLLC) services [5].

To this end, in this work, we are studying the problem of energy-aware latency minimization for multi-domain networks, via dynamic SFC placement for URLLC services with strict performance requirements. We are proposing and developing SCHE2MA, short for Service CHain Energy-Efficient MAnagement. It is an innovative, multi-agent, energy-aware, distributed RL-based, service orchestration framework aiming to minimize service latency in multi-domain B5G networks. The proposed work is decentralized, eliminating any central point of failure and enabling scalability. The contribution of our work is threefold by:

- Jointly minimizing the average service latency and network-wide energy consumption in multiple traffic scenarios by optimizing the placement of service VNFs and minimizing the number of transmissions between servers.
- Demonstrating scalability, parallelization, and decentralization in the SFC placement dynamic decision-making process, by dividing the VNF orchestration between the local technological domains, avoiding that way costly network-wide VNF configurations.
- Evaluate SCHE2MA through model validation and simulation to demonstrate its ability to jointly reduce average service latency by 103.4% and energy consumption by 17.1% compared to existing solutions from the literature.

The remainder of this paper is organized as follows. Section II provides an extensive discussion of the literature and related works. Section III offers an overview of the System Model architecture and introduces the formulation of all network components. Section IV is an in-depth analysis of the problem statement and definition. Section V presents in detail the distributed RL-based modules and operation. Section VI showcases the experimental setup and evaluates the performance of SCHE2MA against other solutions from the literature. Finally, Section VII provides a brief conclusion of this work and our future intentions regarding this work.

II. Literature Overview

Studying and developing energy-aware optimization requires the understanding of SFC management and orchestration as it dynamically transforms network dynamics.

A. Centralized SFC Orchestration

Although a considerable amount of the literature examines the automated SFC placement, the majority of these works consider exclusively centralized solutions. In [6], the authors attempt to maximize the revenue by optimizing the placement of the VNFs, by modeling and solving the problem with Integer Linear Programming (ILP). Similarly, authors in [7] develop a heuristic algorithm to optimize the network bandwidth consumption by taking advantage of the SFC placement.

On the other hand, more recent works study the SFC placement problem using Deep RL (DRL) techniques. They proved to be an important breakthrough and attracted a lot of attention lately as automated decision-making tools. They are able to outperform classical optimization algorithms, and even humans in many industries [8], thanks to lower computational complexity. In [9] the authors use DRL to perform the placement of Virtual Network Function Forwarding Graphs considering the constraints of the underlying infrastructure. Similarly, Pandey et al. in [10] propose EdgeDQN for efficient SFC placement in the edge-cloud to tackle the resource scarcity issue while maintaining low end-to-end delay. They use an hierarchical network model to avoid the expansion of the action space as the network grows over time. In addition, their reward model incorporates energy consumption and the complexity of SFC. They evaluate the results using a simulator and a testbed to demonstrate the proposed algorithm. In [11] and [12], Leoni Santos et al. propose an energy-aware SFC placement algorithm based on Proximal Policy Optimization DRL named Cand-RL. They compare Cand-RL with an Advantage Actor-Critic-based algorithm using simulations based on Brazil's National Teaching and Research Network backbone. Although these works provide satisfactory solutions, they do not consider the complex, multi-domain nature and scale of modern infrastructure. They utilize centralized algorithms that overlook and manage the entire network, limiting scalability and decreasing the performance as the network grows over time, from the number of users or services to the network itself.

B. Multi-Domain SFC Orchestration

Centralized, single-agent algorithms compute the placement of the SFCs in the entire network. It is a costly and non-scalable solution for large networks as it requires enormous problem and actions spaces. In contrast, multi-domain orchestration takes place between multiple independent administrative domains that the SFC is divided upon. In [13], the authors propose an ILP-based solution that addresses the VNF deployment problem in multi-domain networks by reducing the overall energy consumption while maximizing the number of VNFs. Similarly, authors in [14], study the problem of optimal network service deployment across multiple SDN domains with the target of saving energy while achieving the load balancing of multi-domain networks. They demonstrated through simulation that the proposed heuristic service deployment algorithm is efficient and outperforms comparison algorithms in terms of energy consumption and load balancing degree.

Recent works in the literature, study the multi-domain SFC management and orchestration with partially observable DRL techniques. The algorithmic state space can be shared into multiple partially observable states to agents that can solve the issue by cooperating. Authors in [15] and [16] propose a model that tackles the multi-domain SFC placement policy generation by splitting the network into independent divisions with limited visibility of the local infrastructure. They leverage Deep Deterministic Policy Gradient that operates on partially observable states and using rewards that express the quality of the obtained placements using Linear Physical Programming. Through experiments they were able to demonstrate rejection rates of under 2%, cost and latency close to optimal.

C. *Unexploited Gap in the Literature*

We can safely conclude that there is a gap in the literature regarding the energy-aware SFC placement for service latency minimization awaiting to be covered. To the best of our knowledge, we are among the first to propose a distributed RL-based approach for multi-domain SFC orchestration for latency and energy efficiency optimization in this regard.

III. System Model

For our solution, we consider a 5G network consisting of multiple clouds that are geographically distributed in a wide area, as can be seen in Fig 1.

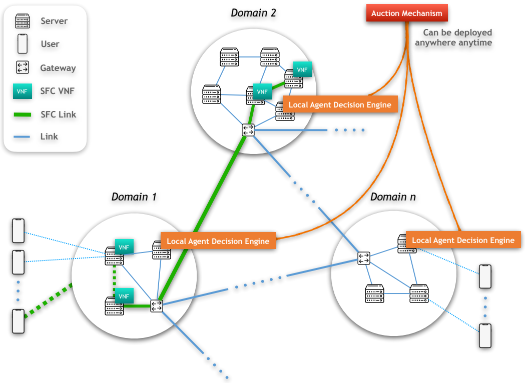

Server
User
Gateway
VNF
SFC VNF
SFC Link
Link
Domain 2
Auction Mechanism
Can be deployed anywhere anytime
VNF
VNF
Local Agent Decision Engine
Domain 1
Domain n
Local Agent Decision Engine
Local Agent Decision Engine
VNF
VNF

Fig. 1. Network graph model.

The geographically distributed clouds are called *Domains*, denoted with n, and consist of interconnected computing servers with computational resources and specific energy consumption, following the standard of modern virtualization NFV software such as OpenStack and Kubernetes. Domains located at the edge of the network provide low latency access to the end-users who generate service requests. Services are comprised of a chain of VNFs called SFC. The need for distributed orchestration is apparent in this setting. The terms service and SFC will be used interchangeably for the rest of this work.

As it is apparent by Fig 1, our proposed algorithm SCHE2MA is designed to operate in a completely decentralized manner. It utilizes the distributed notion of network domains to operate locally. This action enables parallelism, avoids unnecessary VNF migrations between the domains or costly re-orchestration of the entire network. An entity disconnected from the decision function, called the *Auction Mechanism*, is introduced to enable inter-domain VNF migration. SCHE2MA is a distributed decision engine with multiple agents, eliminating a centralized point of failure as the *Auction Mechanism* can be instantiated anywhere in the network.

A. *Network Graph*

The physical network infrastructure is represented as a graph G with nodes and edges:

$$G = (\mathcal{V}, \mathcal{E}, \mathcal{M}, \mathcal{U}), \tag{1}$$

where $\mathcal{V}$ and $\mathcal{E}$ represent the sets of the nodes and links respectively. To accommodate the functions of an SDN-NFV enabled network, $\mathcal{V}$ has two types of nodes. One is dedicated to network switching that is responsible for forwarding the service traffic and the other one has the ability, not only to forward but also to instantiate, terminate or migrate VNFs which are represented by the set $\mathcal{M}$, up to its physical capacity. Both types of nodes handle the routing of the SFC traffic. Finally, $\mathcal{U}$ denotes the set of users. The terms node, server and host, will be used interchangeably.

The parameters $u, v \in \mathcal{V}$ represent two nodes and $uv \in \mathcal{E}$ represents the physical link that connects nodes u and v.

B. *Network Resources*

The network has a finite number of resources that can be measured and tracked through metrics. The following metrics were used to represent the different network resources, in particular:

- The metric C_{uv}^{bw} represents the capacity of a network link uv, whereas w_{uv}^{bw} its utilization ratio.
- C_u^{cpu} the total number of a server u CPU cores, while w_u^{cpu} its utilization ratio.
- C_u^{ram} the total amount of Random Access Memory of a server u, whilst w_u^{ram} its utilization ratio.
- C_u^{hdd} the total amount of storage space of a server u, whereas w_u^{hdd} its utilization ratio.

The instantiation and run-time of a service VNF in the network servers u uses a portion of the aforementioned computational and network resources. These metrics are vital for the decision making process as it will be presented in Section V-B.

C. Service Function Chains

We represent the SFCs as graphs $G_s = (\mathcal{V}_s, \mathcal{E}_s, \mathcal{U}_s)$. The edges of this graph begin from the *source* VNF to the *destination* VNF of the function chain, thus creating a service s. The flow of data between the VNFs has a predefined order. The notation V_s denotes the *source*, the *destination* and *intermediate* servers that the service s traverses through the instantiated VNFs. $u_s, v_s \in \mathcal{E}_s$ represent two nodes in G_s. $\mathcal{E}_s$ indicate the links $u_sv_s \in \mathcal{E}_s$ that connect adjacent VNFs, served by the server nodes u_s and v_s in G_s. Finally, as $\mathcal{U}_s$ we define the set of users that utilize the service s.

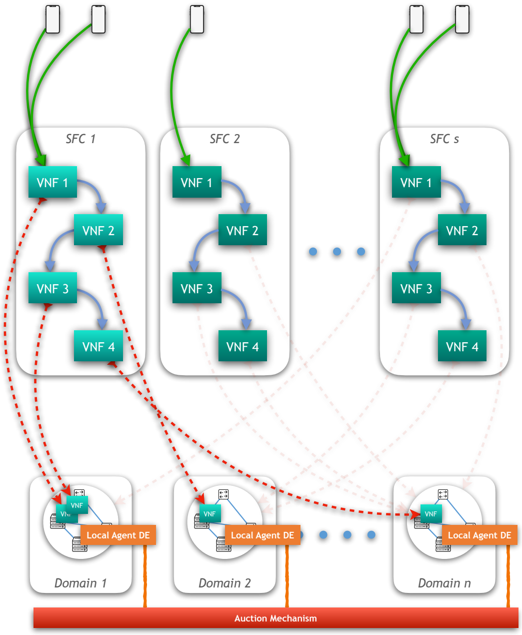


Fig. 2. Multi-domain SFC mapping example.

The dataflow of service s can be also represented as a list $source \rightarrow m_1 \dots m_j \rightarrow destination$, in which $source$ and $destination$ are the *source* and the *destination* nodes of the service. The service traffic need to traverse in between them through the *intermediate* nodes $m_1 \dots m_j$. The traversal of *intermediate* nodes adhere to the *Dijkstra* routing algorithm.

D. Network Resource Constraints

We define services s and categorize them into three distinct classes:

- As $\mathcal{S}$ we describe the instantiated and running services.
- As $\mathcal{T}$ the terminated services.
- As $\mathcal{R}$ the redirected services.

Services can be redirected due to inappropriate VNF placement that violates the user SLA κ_s^{delay} or rejected placement due to insufficient resources to host the service s.

Due to the hardware limitations of the network elements, at any time interval the total allocation of bandwidth and computational resource consumption cannot exceed the available resources in links $\forall uv \in \mathcal{E}$, nodes $\forall u \in \mathcal{V}$ and VNFs $\forall m \in \mathcal{M}$ as described in the following constraints:

$$\sum_{u_sv_s \in \mathcal{E}_s} \sum_{s \in \mathcal{S} \cup \mathcal{T} \cup \mathcal{R}} \phi_s^{bw} z_{uv}^{u_sv_s} \leq \omega_{uv}^{bw} C_{uv}^{bw}, \tag{2}$$

$$\sum_{u_sv_s \in \mathcal{E}_s} \sum_{s \in \mathcal{S} \cup \mathcal{T} \cup \mathcal{R}} \phi_s^{cpu} p_u^m \leq C_u^{cpu}, \tag{3}$$

$$\sum_{u_sv_s \in \mathcal{E}_s} \sum_{s \in \mathcal{S} \cup \mathcal{T} \cup \mathcal{R}} \phi_s^{ram} z_u^{u_sv_s} \leq \omega_u^{ram} C_u^{ram}, \tag{4}$$

$$\sum_{u_sv_s \in \mathcal{E}_s} \sum_{s \in \mathcal{S} \cup \mathcal{T} \cup \mathcal{R}} \phi_s^{hdd} z_u^{u_sv_s} \leq \omega_u^{hdd} C_u^{hdd}, \tag{5}$$

where the variable p_u^m is 1 if m placed in u. Variables $z_{uv}^{u_sv_s}$ and $z_u^{u_sv_s}$ are either 0 or 1 and indicating whether $u_sv_s \in \mathcal{E}_s$ traverses the link uv and server u respectively. The variable ϕ_s^{bw} represents the bandwidth consumption of the physical links due to the traversal of service s. Finally, ϕ_s^{cpu}, $\phi_{s,m}^{ram}$ and ϕ_s^{hdd} are used for the computational resources that the service s VNFs occupy.

E. Maximum Tolerated Delay

The E2E delay of each instantiated and running service $s \in \mathcal{S}$ is highly dependent on its VNF placement. It is calculated as the sum of all link and other network element delays that are traversed by the service chain graph G_s as described in Section III-C. As d_{uv} we denote the delay of link uv, d_u the delay caused by the server hardware and d_m the delay caused by the VNF. If the E2E delay exceeds the maximum tolerated service delay κ_s^{delay}, then an SLA violation is registered. The SLA violation rule is indicated below:

$$\begin{aligned} \sum_{uv \in \mathcal{E}} \sum_{u_sv_s \in \mathcal{E}_s} d_{uv} z_{uv}^{u_sv_s} + \sum_{u \in \mathcal{V}} \sum_{u_sv_s \in \mathcal{E}_s} d_u z_u^{u_sv_s} + \\ \sum_{m \in \mathcal{M}} \sum_{u_s \in \mathcal{V}_s} d_m z_m^{u_s} \leq \kappa_s^{delay} \end{aligned} \tag{6}$$

The inability to achieve the inequality of the equation moves the service $s \in \mathcal{S}$ from operating to redirected $s \in \mathcal{R}$ before the end of the current iteration. Additionally, it triggers a cost penalty used in the reward function to adjust the future actions of the local RL agents, as discussed in a later section.

F. Service Deployment & VNF Orchestration

Fully deployed and operating services $s \in \mathcal{S}$ have a particular way of traversing the network, which cannot be split or reversed. Traffic is flowing from the *source* to the *destination* by traversing all *intermediate* network nodes as described in the following equation:

$$\sum_{u \in \mathcal{V}} \sum_{u_sv_s \in \mathcal{E}_s} (z_{uv}^{u_sv_s} - z_{vu}^{u_sv_s}) = \begin{cases} -1, u \text{ is the Source} \\ 1, u \text{ is the Destination} \\ 0, u \text{ is an Intermediate} \end{cases} \tag{7}$$

If a physical link uv is traversed by the service s, then the nodes connected u and v must be traversed too as follows:

$$z_u^{u_sv_s} z_v^{u_sv_s} = \begin{cases} 1, z_{uv}^{u_sv_s} = 1, \forall u, v \in \mathcal{V}, uv \in \mathcal{E}, u_sv_s \in \mathcal{E}_s \\ 0, \text{otherwise} \end{cases} \tag{8}$$

In addition, the aforementioned links uv and nodes u must be traversed by the service s traffic due to the VNF m placement:

$$\sum_{m\in\mathcal{M}}\sum_{u_s\in\mathcal{V}_s} z_m^{u_s} y_u^m \leq z_{uv}^{u_s v_s}, \forall u_s v_s \in \mathcal{E}_s, \forall s \in \mathcal{S}\cup\mathcal{R}. \quad (9)$$

Finally, (10) ensures that the VNFs m can be placed only in one node u capable of hosting VNFs:

$$\sum_{u\in\mathcal{V}} y_u^m = 1, \forall m \in \mathcal{M} \quad (10)$$

IV. Problem Statement & Definition

In this section, the SFC orchestration will be examined and formulated as an energy-aware, low latency VNF placement problem.

A. Problem Description

Modern network consist of multiple computing domains spread in a wide area. These local computing domains, denoted as n, connect through an infrastructure that we model as a set of inter-domain links uv and servers u. The users $\mathcal{U}_s$ of service s connect to the local domains through a wireless connection, and they request one or more URLLC services from the network. The offered services instantiated as SFCs consisting of multiple VNFs, hosted in various network domains that the traffic flows within. As a result, the network resources are also scattered over a large distances, posing local deprivation of computing resources and increasing the number of hops between the servers needed for the data to travel to complete the service traversal. If there are available resources on the network, it accepts the incoming service requests.

B. Delay Model

The E2E service delay D_s that the service s can offer due to its placement in the network during the current interval can be calculated as follows:

$$D_s = \sum_{uv\in\mathcal{E}}\sum_{u_s v_s\in\mathcal{E}_s} d_{uv} z_{uv}^{u_s v_s} + \sum_{u\in\mathcal{V}}\sum_{u_s v_s\in\mathcal{E}_s} d_u z_u^{u_s v_s} + \sum_{m\in\mathcal{M}}\sum_{u_s\in\mathcal{V}_s} d_m z_m^{u_s} \quad (11)$$

The total service delay $\mathbb{D}$ of the current iteration can be expressed as the average E2E service delay D_s of all services $s \in \mathcal{S}$:

$$\mathbb{D} = \sum_{s\in\mathcal{S}\cup\mathcal{R}} D_s, \quad (12)$$

C. Bandwidth Allocation Model

The total allocated bandwidth B_s of service s occupies from the network link resources due to its placement is calculated as follows:

$$B_s = \sum_{uv\in\mathcal{E}}\sum_{u_s v_s\in\mathcal{E}_s} b_{uv} z_{uv}^{u_s v_s} + \sum_{m\in\mathcal{M}}\sum_{u_s\in\mathcal{V}_s} b_m z_m^{u_s} \quad (13)$$

The average allocated bandwidth $\mathbb{B}$ of the current iteration can be expressed as the average E2E service delay D_s of all services $s \in \mathcal{S}$:

$$\mathbb{B} = \sum_{s\in\mathcal{S}\cup\mathcal{R}} B_s, \quad (14)$$

D. Energy Consumption Model

The total energy consumed by the network during the operation of service s branches into two distinct segments.

First, the energy consumed by the utilization of computational resources E_m while hosting the VNF m on node u is calculated based on the work of Mao, et. al in [17] as defined below:

$$E_m = \phi_s^{cpu}\mu C D_s F^2, \quad (15)$$

where F indicates the computational capacity of the node computing unit measured in CPU cycles per second, C the CPU cycles required for computing one data sample at each CPU core, ϕ_s^{cpu} the number of utilized CPU cores by the VNF and D_s the amount of processed data expressed in bits. The constant μ expresses the effective switched capacitance of the CPU architecture.

Second, the link energy consumption E_{uv} can be calculated as the data sent between the VNFs hosted in server u and v, divided by the data rate and multiplied by the dBm of the link transmitted optical power ρ_{uv}. The transmission energy is calculated as follows:

$$E_{uv} = \rho_{uv}\frac{t_u}{R_u}, \quad (16)$$

where the variable r_u expresses the transmission data rate of server u in Gigabits per second, ρ_{uv} the transmitted optical power of the link uv in dBm and t_u the amount of transmitted data of a server expressed in bits.

The total service $s \in \mathcal{S}$ energy consumption E_s for the operation of the service s in the current iteration is defined as follows:

$$E_s = \sum_{uv\in\mathcal{E}}\sum_{u_s v_s\in\mathcal{E}_s} E_{uv} z_{uv}^{u_s v_s} + \sum_{m\in\mathcal{M}}\sum_{u_s\in\mathcal{V}_s} E_m z_m^{u_s}. \quad (17)$$

The total energy consumption $\mathbb{E}$ of the current iteration can be expressed as the sum of energy consumption E_s of all operating services $s \in \mathcal{S}$:

$$\mathbb{E} = \sum_{s\in\mathcal{S}\cup\mathcal{R}} E_s, \quad (18)$$

E. Problem Objective

In this respect, we formulate the problem of energy-aware low latency SFC management and orchestration as a local long-term constrained optimization task. This enigma translates into solving the local multi-objective constrained task, where we jointly seek to minimize latency while reducing energy consumption and guaranteeing sufficient allocated bandwidth. We formalize the cost function as follows:

$$Minimize\ \mathbb{C} = \frac{w_D * \mathbb{D} + w_E * \mathbb{E}}{w_B * \mathbb{B}}, \quad (19)$$

where w_B, w_D and w_E are the weights used to adjust the gravity of each variable $\mathbb{B}$, $\mathbb{D}$ and $\mathbb{E}$ respectively of the cost function $\mathbb{C}$. It enables fine-tuning to achieve the desired trade-off between allocated bandwidth, latency and energy efficiency.

V. Designing a Distributed Reinforcement Learning solution

This section presents the ZSM energy-aware orchestration problem and proposed solution. The definition of the Distributed RL algorithm and the structure of the proposed *Auction Mechanism*, will be provided.

A. Solution Overview

The ISPs are required not only to maintain but also to improve the service performance and reduce the operating costs by optimizing the placement of VNFs in the network. The division of the network infrastructure, as previously described in the mathematical analysis through equations, in multiple domains remains the main obstacle in resource management optimization due to the high complexity required to be addressed.

In this work, we introduce an intelligent framework to tackle the energy-aware SFC orchestration problem in multi-domain networks called SCHE2MA. It employs multiple RL agents instantiated in each domain n that perform VNF orchestration locally and provides a system for inter-domain migration, offering a *quid pro quo* between local and global SFC orchestration benefits. We build SCHE2MA based on the defined network infrastructure system model presented in the previous section.

B. Markov Decision Process Environment

The intra-domain VNF placement and orchestration problem is formalized as a Markov Decision Process (MDP) which consists of a *State*, an *Action Space* and a *Reward Function*. The problem space of a domain n is defined as follows:

1) State Space: Consists of the domain n computational resources of the domain nodes u that host the VNFs m of the services $s \in \mathcal{S}$. A local domain state S_n is defined as a set where each variable represents the user SLAs and VNF currently in auction requirements of each domain n utilization ratio:

$$\begin{aligned} S_n = (w_{uv}^{bw}, w_u^{cpu}, w_u^{ram}, w_u^{hdd}, \\ \dots, \kappa_s^{delay}, \phi_s^{bw}, \phi_{s,m}^{cpu}, \phi_{s,m}^{ram}, \phi_{s,m}^{hdd})^T, \end{aligned} \tag{20}$$

2) Action Space: The *Confidence Vector* A_n, a vector that contains a bid $\in [-1, 1]$ for each domain server u_n to receive and host the VNF currently in auction by the *Auction Mechanism*. The maximum of this local action indicates an internal VNF migration:

$$A_n = (i_0, \dots, i_{u_n})^T, \tag{21}$$

3) Reward: Consists of cost function expressed in (22). The objective of all domains n is to maximize the sum of function R^{shared} in every iteration and thus, converging in a common state-action that enables cooperation among the domains:

$$R^{shared} = \frac{w_B * \mathbb{B}}{w_D * \mathbb{D} + w_E * \mathbb{E}} + P, \tag{22}$$

In addition, we define a penalty function that is applied when the SLA described in (6) of the service s is violated:

$$P = -w_P(D_s > \kappa_s^{delay}) \tag{23}$$

The penalty inclusion assists the algorithm to converge faster by accumulating massive constraint costs during the exploration phase of the distributed RL agents.

C. Distributed Reinforcement Learning Agents Structure

The problem of this work includes multiple MDP environments with local RL agents that are located in all domains n. The environments consist of the local computational and network resources of the domain n as defined in the previous section.

To build the domain n RL agents, we leverage Deep Q-Network agent learning as defined by the work of Mnih et al. in [18]. The goal of the agents is to select an action as defined in 21, VNF placements in this case, at every state 20 that maximizes the accumulated reward, as described in (22). We use a Deep Neural Network (DNN) to approximate the optimal action-value function, also known as Q-value function:

$$Q^*(s, a) = \max_{\pi} E[r_t + \gamma_{t+1} + \gamma^2 r_{t+2} + \dots | s_t = s, a_t = a, \pi], \tag{24}$$

where the Q-value function can be defined as the maximum sum of all rewards r_t, discounted by the parameter γ at each time-step t. The maximum sum of all rewards r_t is achieved by a behavioral policy $\pi = P(a|s)$, after an *Observation* s and taking an *Action* a.

To avoid instability during the training of the agents, we employ the *Experience Replay* technique, which randomizes the *Observations* and removes the correlation between them during the early training phase to force the agent to embrace exploration. We store the experiences $e_t = (s_t, a_t, r_t, s_{t+1})$ of the agent at each time-step t in the set $D_t = e_t, \dots, e_t$, that we later use to retrieve them. We apply Q-learning value updates on mini-batches of experience (s, a, r, s'), $U(D)$, drawn uniformly at random from the set D_t of stored experiences to perform learning for the agent. The Q-learning update during iteration i utilizes the loss function $L_i(\theta_i)$, which is defined as following:

$$\begin{aligned} L_i(\theta_i) = \epsilon_{(s,a,r,s')\ U(D)}[(r + \gamma \max_{a'} Q(s', a'; \theta_i^-) \\ -Q(s, a; \theta_i))^2], \end{aligned} \tag{25}$$

where γ denotes the discount factor that determines the agent's horizon, θ represents the parameters of the Q-network during iteration i and θ_i^- are the network parameters used to compute the target value at iteration i.

The *Confidence* metric which represents the domain agent *bid* during the *Auction Mechanism* auction procedure is obtained after applying the *arguments of the maxima*, also known as the *argmax* function, which is described with the following equation:

$$\underset{x \in D}{\text{argmax}} f(x) = \{x | f(x) \geq f(y) \forall y \in D\}, \tag{26}$$

where $f(x)$ is the set of inputs x from the DNN output D that achieve the highest function value. The *Confidence* metric *bid* is extracted from the set D as $max(D)$, whereas $f(x)$ denotes the intra-domain placement, pointing at the server with the highest *Confidence*.

D. Auction Mechanism Architecture

In this work, we also introduce the *Auction Mechanism*, a system that enables inter-domain VNF migration in a distributed multi-domain network. As shown in Fig. 3, the *Auction Mechanism* enables scalability and parallel operation.

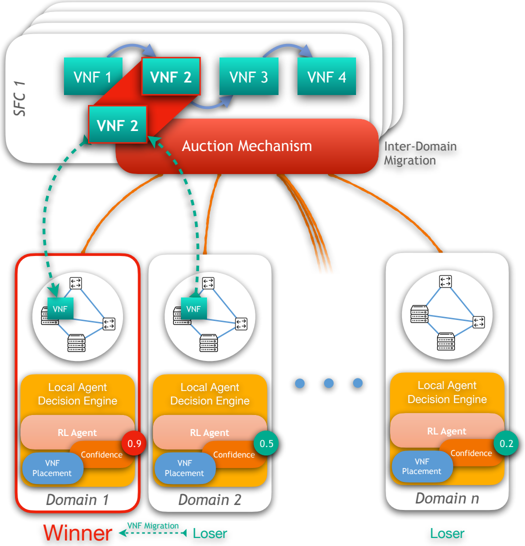


Fig. 3. Overview of the multi-domain and distributed Auction Mechanism. All local domains can orchestrate VNFs in parallel and exchange VNFs only during an auction.

The operation of the *Auction Mechanism* can be described in the following steps:

1) Auction Initiation: The *Auction Mechanism* chooses the next service s VNF m and showcases to the distributed domains n the requirements of the placement.

2) Distributed Operation: The distributed RL agents of the domains n generate their local action A_n or *Confidence Vector* to propose a local placement for the showcased VNF. The $argmax$ function *Confidence Vector* the *Confidence Metric* of each domain is sent to the *Auction Mechanism*, ensuring minimum data transfers.

3) Global Operation: The *Auction Mechanism* receives the *Confidence Metric* of each domain and selects the *highest bidder* or the domain with the maximum *Confidence Metric* as a candidate to receive the VNF currently in auction. The *Auction Mechanism* notifies the candidate domain with an acknowledgment response.

4) Orchestration: If the candidate domain is different from the current domain that hosts the VNF in the auction, the inter-domain migration is initiated. Contrariwise, the domain agent performs an intra-domain migration to the node with the highest *Confidence Metric* of the local *Confidence Vector* with a much lower cost in terms of both energy, time and overall cost. If the VNF is already instantiated in the same node, the procedure of migration is declined.

5) Iteration: The procedure is repeated indefinitely.

It is apparent that the *Auction Mechanism* acts as the *auctioneer* and it is only responsible for the inter-domain communication making it non-essential for the local domain orchestration. The *Auction Mechanism* can be deployed quickly at any node of the network eliminating the single point of failure in the system.

VI. Simulation Results & Evaluation

In this section, we conduct a simulation study with a diverse variety of scenarios on a realistic multi-domain network to prove the performance superiority of the proposed scheme.

A. Simulation Environment

The equations and models presented in the previous sections were simulated with *Python* and a custom-made OpenAI Gym environment [19] was used to make it accessible to the RL agents. The RL agents were developed using *TensorFlow* [20] and the high-level *Keras API* open-source library [21]. The network environment is a fork of Containernet [22], an advanced branch of Mininet [23] network emulator used for evaluation by many works in related literature. It simulates a realistic virtual network, VM hosting, switching, and application code for developing and experimenting with SDN-NFV networks. The simulated network topology is a variation of the *2005 Nordu European network*, from *The Internet Topology Zoo* [24], adjusted to accommodate multiple computational domains and fit the requirements of the study. Each node of the topology corresponds to a domain and more nodes are introduced according to the scale of the experiment. The initial and non-variable simulation parameters are presented in Table I. The discreet sets signify multiple value options to accommodate the experiments, whereas ranges indicate a randomly selected value from the given set.

TABLE I
Simulation Parameters

Users	
Number of users	$\mathcal{U} \in \{100, 500, 1000, 1500, 2000\}$
Packet loss probability	$\psi_y^{loss} \in [2\%, 5\%]$
Domains	
Number of domains	$n \in \{3, 5, 7, 9, 11\}$
Servers per domain	$u_n = 3$
Service Function Chains	
Number of services	$\mathcal{S} \in 25, 125, 250, 375, 500$
CPU requirement	$\phi_s^{cpu} \in [1, 2]$
RAM requirement	$\phi_s^{ram} \in [2, 4]$ GB
Storage requirement	$\phi_s^{hdd} \in [8, 64]$ GB
Service latency SLA	$\kappa_s^{delay} \in [5, 15]$ ms
Servers	
CPU capacity	$C_u^{cpu} = 64$
Memory capacity	$C_u^{ram} = 128$ GB
Storage capacity	$C_u^{hdd} = 1$ TB
Links	
Intra-domain link latency	$d_{uv} \in [1, 2]$
Inter-domain link latency	$d_{uv}^{wan} \in [2, 5]$

B. Baseline Scenarios

The performance of the proposed SCHE2MA solution is compared with two references from the literature scenarios:

1) Centralized RL: An RL-based orchestration algorithm located in a central location, a common type of baseline approach in the related research literature, such as in [6], [7], and [9]. The central orchestration algorithm overlooks the entire network as opposed to our proposed distributed orchestration scheme, the VNFs are serially placed and the VNFs are migrated to the node with the highest action value.

2) Static Placement: A typical VNF placement strategy, which is adopted by many providers even today, as the default baseline [25]. In this strategy the VNF placement is static and the VNFs remain hosted in the initial node throughout the experiment.

C. Results Analysis

The performance of the baseline scenarios is normalized to the SCHE2MA performance, and the plots show the relative gain or loss for each metric. The analysis shows that the performance of SCHE2MA in both average energy consumption and average service latency. The energy consumption curves of all figures are normalized based on the SCHE2MA performance to improve legibility. The values are expressed in millijoules (mJ) under the curve of SCHE2MA.

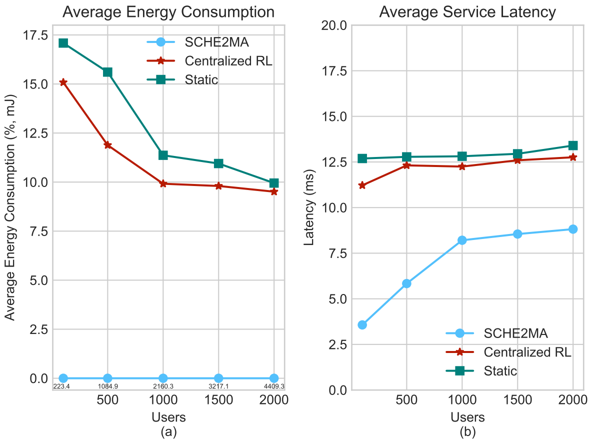


Fig. 4. (a) Average energy consumption improvement expressed in percentages, compared to the proposed solution. (b) Average service latency per number of users in multiple traffic scenarios. Lower is better for both figures.

In Fig. 4a, we depict the average energy consumption of the examined network of 500 simulations for a varying number of users, normalized based on the SCHE2MA performance (%,mJ). We observe that the energy consumption increases almost linearly with the number of users due to the massive number of transmissions. The reason is that introducing more users to the network generates additional requests that consume more energy during each transmission. Therefore, the overall energy consumption of the network is higher. It is evident that the proposed solution is able to maintain lower energy consumption in all scenarios, reaching almost 17.1% reduction in the case of 100 users. The reason for this behavior is the ability of SCHE2MA to cluster VNFs into the servers, minimizing the costly communication between servers.

Fig. 4b presents the performance of the most critical metric in URLLC services, the average service latency. We observe that the average service latency increases due to insufficient computing resources in servers within the domains as the number of active users grow. However, it has to be noted that SCHE2MA outperforms both baselines by offering a 103.4% reduction in latency for the case of 100 users without increasing the energy consumption, which is a considerable performance improvement while also maintaining lower energy consumption than both baselines. That is possible due to VNF clustering in servers, which minimizes the number of transmissions in physical media. SCHE2MA demonstrates a clear indication of its ability to conceive better VNF placements that satisfy the latency and energy consumption trade-off.

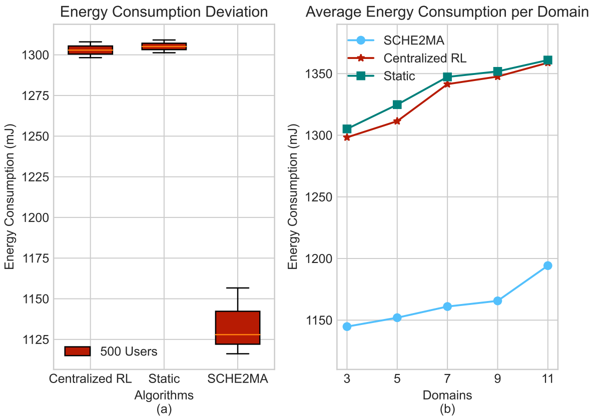


Fig. 5. (a) Energy consumption deviation in a 3 domain network with 500 users and 25 SFCs. (b) Average energy consumption in multi-domain network configurations for 500 users and 25 SFCs. Lower is better for both figures.

Fig. 5a presents how the energy consumption fluctuates during the operation of each algorithm, specifically for the scenario of 3 domains and 500 users within a simulation cycle. We observe that the maximum difference in energy consumption is 15.91% between the Static solution and SCHE2MA. The reason for this is that, as can be seen in Fig. 8a SCHE2MA tends to consolidate multiple SFC VNFs in hosts id-est hosts 2 and 5, to minimize both energy consumption and latency by turning physical link connections into virtual that yield minimal losses. In Fig. 5b, we plot the average energy consumption per domain in order to evaluate the scalability of the given solutions. We observe that the energy consumption steadily increases as we introduce more domains into the network, hence increasing the number of data that need to be considered when planning a VNF placement. It can be seen that the Centralized RL fails to converge due to the larger state space. SCHE2MA is able to reduce the energy consumption by 14.85% compared to the baseline solutions comparing to scenarios id-est the 9 domain network with 500 users. This behavior is due to the flexibility and scalability of SCHE2MA's distributed architecture where the decision-making takes place locally in multi-domain agents that communicate through the *Auction Mechanism*, dividing and sharing that way the immense problem space.

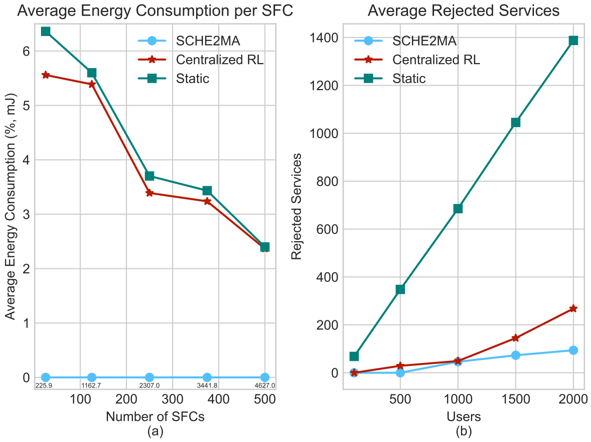


Fig. 6. (a) Average energy consumption by the number of SFCs in the network for 500 users. (b) Average number of rejected services for a 3 domain network. Lower is better for both figures.

Fig. 6a outlines the average energy consumption per SFC deployed in the network. We observe that in the case of 25 SFCs, the average energy consumption per SFC of SCHE2MA is reduced by 6.36% compared to the *Static* solution. The reason is that compared to the baseline scenarios, SCHE2MA is capable of operating with less energy, as we have previously discussed and analyzed in Fig. 4. Fig. 6b illustrates the average number of rejected services in a 3 domain scenario with a varying number of users. When the number of users increases in a network with finite resources, the number of rejected services increases. Given that the SCE2MA can re-organize the VNFs, a number of resources can be released. We can conclude that the improvements can be attributed to the VNF consolidation abilities of SCHE2MA.

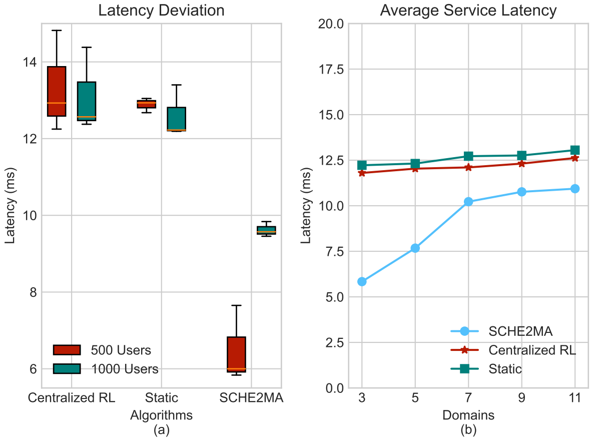


Fig. 7. (a) Latency deviation per algorithm for both 500 and 1000 user traffic scenarios in a 3 domain network. (b) Average service latency of 500 users in a 3 domain network. Lower is better for both figures.

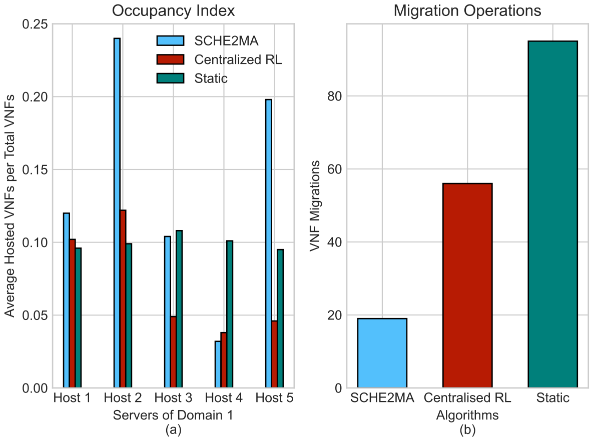


Fig. 8. (a) VNF occupancy index represents the average number of hosted VNFs per total number of VNFs for 100 iterations of simulated traffic with 500 users. (b) Average migration operations of 100 iterations of simulated traffic with 50 SFCs and 125 VNFs.

Finally, Fig. 7a illustrates how the service latency oscillates during the operation of each algorithm for the scenarios with 500 and 1000 users. We observe that SCHE2MA was able to achieve 73.52% less service latency than the baseline scenarios in the case of 5 domains, depicted in Fig. 7b. That is possible by devising VNF placements that minimize the number of transmissions through local intra-domain orchestration. The *Centralized RL* is hugely affected by the number of users, as the deviation in the figure suggests. Fig. 8a shows the average number of hosted VNFs divided by the number of total service VNFs to indicate the *occupancy* of the hosts of the the first domain. We can see that SCHE2MA gravitated towards consolidating the SFC VNFs to reduce the number of hops to the end-user. Additionally, Fig. 8b illustrates the total number of migrations of the local agent originally depicted in Fig. 8a that was applying an identical placement for a sustained period to avoid inter-domain SFC re-configurations and additional data transmissions that lead to higher energy consumption and latency.

VII. Conclusion & Future Work

In this work, we have studied the problem of energy-aware latency minimization for multi-domain networks, via dynamic SFC placement for URLLC services. We have proposed SCHE2MA, an innovative, multi-agent and distributed RL-based service orchestration framework. We have introduced the *Auction Mechanism* that the local domains use to exchange VNFs between the domains. The results confirm superior performance in multiple scenarios, maintaining the high levels of efficiency in multi-domains scenarios compared to a *Centralized RL* agent solution.

As future work, we intend to implement the *Auction Mechanism* in a totally decentralized way by employing *Blockchain* technology. The domain agents will place bids in the new auction process through *Smart Contracts* and a distributed and immutable public ledger.

Acknowledgment

Supported by the EC projects 5GMediaHUB (101016714), 5G-ROUTES (951867), MARSAL (101017171), OPTIMIST (872866) and MonB5G (871780).

References

[1] J. Gil Herrera and J. F. Botero, "Resource allocation in nfv: A comprehensive survey," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 518–532, 2016.

[2] Y. Goto, "Standardization of automation technology for network slice management by etsi zero touch network and service management industry specification group (zsm isg)," *NTT Technical Review*, vol. 16, pp. 39–43, 09 2018.

[3] R. Mijumbi, J. Serrat, J.-l. Gorricho, S. Latre, M. Charalambides, and D. Lopez, "Management and orchestration challenges in network functions virtualization," *IEEE Communications Magazine*, vol. 54, no. 1, pp. 98–105, 2016.

[4] A. Dalgkitsis, P.-V. Mekikis, A. Antonopoulos, G. Kormentzas, and C. Verikoukis, "Dynamic resource aware vnf placement with deep reinforcement learning for 5g networks," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

[5] A. Dalgkitsis, L. A. Garrido, P.-V. Mekikis, K. Ramantas, L. Alonso, and C. Verikoukis, "Schema: Service chain elastic management with distributed reinforcement learning," in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 01–06.

[6] 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, cloud Networking and Communications II.

[7] Z. Ye, X. Cao, J. Wang, H. Yu, and C. Qiao, "Joint topology design and mapping of service function chains for efficient, scalable, and reliable network functions virtualization," *IEEE Network*, vol. 30, no. 3, pp. 81–87, 2016.

[8] D. Silver *et al.*, "Mastering the game of go without human knowledge," *Nature*, vol. 550, pp. 354–359, 10 2017.

[9] P. T. A. Quang, Y. Hadjadj-Aoul, and A. Outtagarts, "A deep reinforcement learning approach for vnf forwarding graph embedding," *IEEE Transactions on Network and Service Management*, vol. 16, no. 4, pp. 1318–1331, 2019.

[10] S. Pandey, T. Van Nguyen, J.-H. Yoo, and J. W.-K. Hong, "Edgedqn: Multiple sfc placement in edge computing environment," in *2021 17th International Conference on Network and Service Management (CNSM)*, 2021, pp. 301–309.

[11] G. L. Santos, T. Lynn, J. Kelner, and P. T. Endo, “Availability-aware and energy-aware dynamic sfc placement using reinforcement learning,” *The Journal of Supercomputing*, vol. 77, no. 11, pp. 12 711–12 740, Nov 2021.

[12] G. L. Santos, P. T. Endo, T. G. Lynn, D. H. J. Sadok, and J. Kelner, “A reinforcement learning-based approach for availability-aware service function chain placement in large-scale networks,” 2022.

[13] K. Kaur, S. Garg, G. Kaddoum, F. Gagnon, N. Kumar, and S. H. Ahmed, “An energy-driven network function virtualization for multi-domain software defined networks,” in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2019, pp. 121–126.

[14] C. Zhang, X. Wang, A. Dong, Y. Zhao, Q. He, and M. Huang, “Energy efficient network service deployment across multiple sdn domains,” *Computer Communications*, vol. 151, pp. 449–462, 2020.

[15] N. Toumi, M. Bagaa, and A. Ksentini, “On using deep reinforcement learning for multi-domain sfc placement,” in *2021 IEEE Global Communications Conference (GLOBECOM)*, 2021, pp. 1–6.

[16] N. Toumi, O. Bernier, D.-E. Meddour, and A. Ksentini, “On cross-domain service function chain orchestration: An architectural framework,” *Computer Networks*, vol. 187, p. 107806, 2021.

[17] Y. Mao, J. Zhang, and K. B. Letaief, “Dynamic computation offloading for mobile-edge computing with energy harvesting devices,” *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3590–3605, 2016.

[18] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, “Human-level control through deep reinforcement learning,” *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[19] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, “Openai gym,” 2016.

[20] M. Abadi *et al.*, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015.

[21] F. Chollet *et al.*, “Keras,” https://keras.io, 2015.

[22] M. Peuster *et al.*, “Medicine: Rapid prototyping of production-ready network services in multi-pop environments,” in *2016 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, Nov 2016, pp. 148–153.

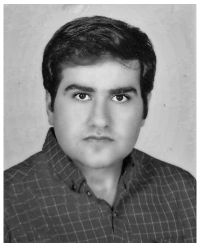
[23] B. Lantz *et al.*, “A network in a laptop: Rapid prototyping for software-defined networks,” 01 2010, p. 19.

[24] S. Knight, H. Nguyen, N. Falkner, R. Bowden, and M. Roughan, “The internet topology zoo,” *Selected Areas in Communications, IEEE Journal on*, vol. 29, no. 9, pp. 1765 –1775, october 2011.

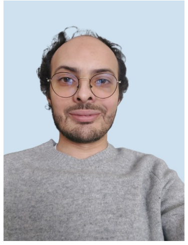
[25] A. Fischer, J. F. Botero, M. T. Beck, H. de Meer, and X. Hesselbach, “Virtual network embedding: A survey,” *IEEE Communications Surveys Tutorials*, vol. 15, no. 4, pp. 1888–1906, 2013.

Anestis Dalgkitsis is currently a Ph.D. candidate at the Polytechnic University of Catalonia (UPC), a visiting researcher at Ericsson Ireland and a Research Engineer at Iquadrat Informatica S.L. His work mainly focus on AI-based zero-touch slice management and orchestration for 5G/6G networks, and specifically for ultra-low latency services. He is involved in multiple European Commission funded H2020 projects. He holds a M.Eng. degree in Informatics & Telecommunications Engineering from University of Western Macedonia, Greece.

Luis A. Garrido is a Senior Researcher at Iquadrat Informatica S.L since March 2020. He has received his PhD degree from the Department of Computer Architecture of the Polytechnic University of Catalonia (UPC), Spain, in 2019. Also, he holds an Electronical Engineering and Telecommunications degree (2011) and a M.Sc. in Electrical Engineering and Computer Science (2013) from Technological University of Panama and National Chiao Tung University (NCTU), Taiwan, respectively. Currently, he is actively involved in EU-funded research projects, while his main research interests include Low-Level Virtualization, Cloud Computing, Machine Learning, and 5G/6G Communication Networks.



Farhad Rezazadeh (S’19) is currently a Ph.D. candidate at the Technical University of Catalonia (UPC) and Researcher at CTTC. He is involved in some European H2020 projects and he was awarded first patent connected to H2020 5G-SOLUTIONS project. He was a secondee at NEC Lab Europe and had scientific mission at TUM, TUHH, and UdG. He won multiple European, Government, and IEEE ComSoc Fellowship/Grants. He also serves/served as Reviewer, Organizing, and TPC member in IEEE. His research interests lie in the area of Lifelong Machine Learning, B5G/6G, and Network Slicing.



Hatim Chergui (M’12) received the engineering degree in telecommunications from the Institut National des Postes et Télécommunications (INPT), Rabat, Morocco, in 2007 and the Ph.D. degree (summa cum laude) in electrical engineering and telecommunications from IMT-Atlantique (Télécom-Bretagne), Brest, France, in 2015. He is currently the project manager of the H2020 MonB5G European project and a researcher at CTTC, Spain. He served as a RAN expert at both INWI and Huawei Technologies, Morocco. He was the recipient of the IEEE ComSoc CSIM 2021 Best Journal Paper Award and the IEEE ICC 2020 Best Paper Award. He is an Associate Editor of IEEE Networking Letters.

Kostas Ramantas has received the Diploma of Computer Engineering, the MSc degree in Computer Science and the PhD degree from the University of Patras, Greece, in 2006, 2008 and 2012 respectively. He has been the recipient of two national scholarships. In June 2013, he joined IQUADRAT as a senior researcher and has co-supervised 1 PhD student, 2 ESRs in ITNs and many secondments in the framework of RISE projects (e.g., CASPER, WATER4CITIES) and ITN projects (e.g., Spotlight, 5GAura). He was involved in multiple E.C. funded projects (e.g. IoSENSE), while he is currently the TM in 5GMediaHUB project. He has published more than 35 journal and conference papers. His research interests are in modeling and simulation of network protocols, and scheduling algorithms for QoS provisioning.

John S. Vardakas (Senior Member, IEEE) received the Dipl. Eng. in Electrical Computer Engineering from the Democritus University of Thrace, Greece, in 2004 and his Ph.D from the Electrical Computer Engineering Dept., University of Patras, Greece in 2012. He has authored more than 40 journal articles and 70 conference articles, while he has participated in more than 15 competitive research programs. His research interests include teletraffic engineering, performance analysis and simulation of communication networks and smart grids. He is a senior member of the IEEE and a member of the Technical Chamber of Greece (TEE).

Dr. Christos Verikoukis received the Ph.D. degree from Technical University of Catalonia (UPC), Barcelona, Spain, in 2000. He is currently a scientific director in Iquadrat Informatica, and an Associate Proferssor with the Univerisity of Patras. He has authored 151 journal papers and over 200 conference papers. He is also a coauthor of three books, 14 chapters in other books, and two patents. He has participated in more than 40 competitive projects, and has served as a principal investigator of national projects in Greece and Spain. He is currently the IEEE ComSoc GITC vice-chair and the editor-in-chief of the IEEE Networking Letters.