Service Function Chain Placement in Distributed Scenarios: A Systematic Review

Guto Leoni Santos1 · Diego de Freitas Bezerra1 · Élisson da Silva Rocha1 · Leylane Ferreira1 · André Luís Cavalcanti Moreira1 · Glauco Estácio Gonçalves2 · Maria Valéria Marquezini3 · Ákos Recse4 · Amardeep Mehta5 · Judith Kelner1 · Djamel Sadok1 · Patricia Takako Endo6

Received: 22 December 2020 / Revised: 7 June 2021 / Accepted: 26 August 2021 / Published online: 14 September 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract
The network function virtualization (NFV) paradigm is an emerging technology that provides network flexibility by allowing the allocation of network functions over commodity hardware, like legacy servers in an IT infrastructure. In comparison with traditional network functions, implemented by dedicated hardware, the use of NFV reduces the operating and capital expenses and improves service deployment. In some scenarios, a complete network service can be composed of several functions, following a specific order, known as a service function chain (SFC). SFC placement is a complex task, already proved to be NP-hard. Moreover, in highly distributed scenarios, the network performance can also be impacted by other factors, such as traffic oscillations and high delays. Therefore, a given SFC placement strategy must be carefully developed to meet the network operator service constraints. In this paper, we present a systematic review of SFC placement advances in distributed scenarios. Differently from the current literature, we examine works over the last 10 years which addressed this problem while focusing on distributed scenarios. We then discuss the main scenarios where SFC placement has been deployed, as well as the several techniques used to create the placement strategies. We also present the main goals considered to create SFC placement strategies and highlight the metrics used to evaluate them.

Keywords Network function virtualization · Service function chain · Distributed scenarios · Systematic review

Patricia Takako Endo
patricia.endo@upe.br
Extended author information available on the last page of the article
1 Introduction

With the rise of 5G and Internet of Things (IoT) technologies, it is expected that the network traffic and the number of active devices increase dramatically over the next few coming years [1]. In this context, different applications with different network requirements will demand customized network configurations [2]. Additionally, guaranteeing Service Level Agreements (SLA) via automation in a highly dynamic environment—as in the 5G and IoT scenarios where the number of connected users, services and data traffic vary over time—represents a considerable challenge.

Traditionally, network functions (e.g. proxies, network address translation, and firewalls) are offered as proprietary closed hardware-driven implementations or dedicated middle-boxes [3]. Each vendor provides its own management software and firmware, resulting in significant capital investment to create and launch a new service [4]. This lack of flexibility in traditional networks turns their integration and deployment of new service cumbersome while compromising network management since middle-boxes have expensive and slow provisioning cycles besides facing waste of resources in low traffic due to difficult re-purposed and must be dimensioned at peak loads [5, 6].

NFV promises virtual services deployment in a more cost-efficient and flexible manner [5]. Under such a paradigm, network functions are decoupled from the physical network and deployed on low-cost commodity hardware using virtualization (running on virtual machines or simpler containers) [7]. In addition, the Software Defined Networking (SDN) paradigm suggests the separation of control and data planes when managing network flows. Despite their separate development paths, the integration between NFV and SDN can increase flexibility in terms of network abstraction, infrastructure sharing, and reconfigurability [8]. For instance, SDN can dynamically be used to control network flows and routes based on criteria that could involve, e.g, network traffic condition [5] and control load.

When creating a new service based on NFVs, there are some cases in which the network manager needs to specify the service functions and their respective execution order (or part of them). This way, a Service Function Chain (SFC) is created to formally represent such service. A SFC is composed of a set of VNFs and virtual links that connect them, where the VNFs have an execution order [9]. It is also worth that different SFCs from different tenants may share the same hardware components (physical machines and network links) to improve resource utilization and hence operating cost.

SFC orchestration consists of determining “a feasible path on the physical network where the vertexes and edges of the path can satisfy both the computing resource requests of VNFs and the bandwidth demands of virtual links” [2], but there are also cases such as multi-domain SFC networks, where the VNFs need to be deployed in different domains through non-linear paths [10]. However, SFC orchestration can be a complex task. For example, considering a large data center hosting hundreds of racks with thousands of servers and dozens of network components [11], allocating the VNFs and maximizing resource usage in this kind of complex scenario is quite challenging. As demonstrated by Kuo et al. [12], SFC mapping is
an NP-hard problem, in which obtaining an optimal solution is often difficult and time-consuming.

Furthermore, additional challenges emerge when a distributed scenario is considered. Different from centralized scenarios, when for instance, a distributed cloud architecture is considered, the unpredictable communication delay can become a high risk factor for the time sensitive applications [13]. Indeed, considering a multi-domain scenario, the communication over the Internet can make it difficult to meet delay constraints [14], and the SFC placement solutions need to consider the high delay of links. In addition, the links that connect different domains and/or data centers may have different costs, and particularity need to be considered in the SFC placement solution in order to find the optimal placement strategy [15]. Liu et al. [16] also highlighted that the SFC placement in distributed scenarios in comparison to centralized scenarios brings additional challenges such as confidentiality of intra-domain information, domain’s local autonomy, and hardware variety. These aspects must be considered in SFC placement solutions in order to ensure the network manager goals meeting its constraints.

Another challenge faced in SFC context regards to availability in these networks. As noted by Fan et al. [9], even when the availability of each virtual function is high, the availability of the whole SFC may not be acceptable. For instance, considering a linear SFC composed of three VNFs, where each one has an availability equals to 95%, the availability of the whole SFC is 85% approximately (the multiplication of the availability of all VNFs). This can become a problem for operators as different network and link events may impact SFC availability, including network downtime, software failures, misconfigurations, and infrastructure failures [4].

This paper presents a systematic review of SFC placement in distributed scenarios, discussing the main work contributions in the last decade. We evaluated the SFC placement goals, the solutions proposed for the SFC placement, and the scenarios considered in the papers. We also present which metrics are considered in the papers to evaluate their solutions. Finally, we discuss the results found in this systematic review.

2 Related Work

Bhamare et al. [17] presented a survey of architectural approaches focused on the implementation and automation of SFC in multi-cloud environments. Authors classified the works into (a) architectural models and implementation, and (b) optimization models. The authors highlighted that despite the existence of a significant amount of research on SFC, major issues remain open for consideration, including the need for the definition of a stable and reliable SFC architecture.

A recent work from Bonfim et al. [8] presented a systematic literature review with particular emphasis on integrated NFV and SDN architectures that support SFC allocation in distributed cloud scenarios. The authors investigated the state of the art of NFV and SDN architectures and identify research topics for further improvements. Authors argued that current architectures for Mobile Edge Computing (MEC)
are still immature, and better approaches are necessary to offer an efficient distributed SFC management.

The survey presented by Rotsos et al. [18] considers existing standardization efforts for the orchestration (that includes automation, coordination, and management processes) of SFCs from an operator’s perspective. Authors discussed some industry challenges for the standardization; they argued that some protocol solutions are still in their infancy and require further investigation and development before they can be successfully deployed in complex and large-scale environments.

In [19], Hantouti et al. presented a survey of traffic steering for SFC, analyzing its mechanisms on the efficiency of service chaining. Authors classified the studies into three approaches: (a) based on programmable switches, it means SDN switches (b) based on switches with compact packet tags and (c) based on packet header. Based on these classifications, they analyzed a list of SDN-based SFC methods used to traffic steering. Furthermore, authors presented a qualitative evaluation of SDN-based SFC approaches, analyzing the configuration used in these studies.

In [20], Souza et al. systematically reviewed the integration of NFV in data centers. They highlighted some research problems related to performance, dependability, resource allocation, cost, management, and resource interoperability. The authors summarized the main lessons learned from the literature, and also discuss some open research challenges. They stated that “in the case of SFCs, new challenges appear because they have to deal with the implementation of multi-vendor service functions in geographically distributed data centers”.

In [21], Schardong et al. carried out a systematic review and proposed a taxonomy for NFV resource allocation. They focused specifically on the VNF Forwarding Graph Embedding (VNF-FGE) problem. The studies were classified based on the proposed taxonomies, consisting of: Behavior (online or offline), Allocation task (Placement, Chaining or both), Objective (reduce network resources or host resources or economic cost, or trade-off between network and host resources), Factors (storage, memory, link capacity or latency, among others) and technique style (exact, heuristic or meta-heuristic). The authors also identified challenges and deficiencies in the development of (sub-)optimal solutions for the problem in question, such as life cycle management of VNFs and comparison of proposed algorithms.

Our systematic review complements the works done by [20, 21], highlighting the existing solutions for SFC placement in distributed scenarios proposed over the last decade. Furthermore, in comparison with other surveys such as [8, 17], our systematic review provides a more comprehensive view, presenting SFC placement goals and solutions as well as the scenarios in which the solutions are applied. Finally, our work does not focus on specific metrics as presented in [18, 19].

3 SFC Definitions

As mentioned by Mirjalily et al. [22], there are four stages needed to deliver a SFC. The following stages are identified for handling possible issues related to automatic SFC deployment in network infrastructures: description, composition, placement,
and scheduling, as illustrated in Fig. 1. A generic SFC example is considered in Fig. 1 composed of the three functions VNF1, VNF2, and VNF3.

The first stage is the description which is responsible for describing each VNF and detailing its functional (e.g., type of service provided) and nonfunctional (e.g., information about disk, memory, number of CPUs) properties. This stage is important for network providers to specify the functionalities and requirements of each VNF needed to make the SFC completely operational [22]. An example of SFC definition is shown in Fig. 2. There are two physical nodes represented by source and destination blue boxes. From the source, there are three chained VNFs (magenta, red, and purple boxes) with different requirements in terms of CPU, memory, and storage, to achieve the destination. The source, VNFs and destination entities are connected through links with specific bandwidth requirements. Although, we use simple cloud resource requirements for the sake of this example, in real scenarios there is a much broader set of possible requirements (e.g. affinity, anti-affinity, hardware acceleration).

Several alternatives can be applied for the VNF description. Both the Web Service Description Language (WSDL) [23] and the Unified Service Description Language (USDL) [24, 25] are efforts for web services description, while Cloud Service Declarative Definition Language (CSDL) and its variations [26] focus on the description of cloud services. As highlighted by Mehraghdam and Karl [27], the YANG data model can be used for the VNF description. An additional data model that can be used is the Topology and Orchestration Specification for Cloud Applications (TOSCA), which
provides a data model and templates to orchestrate and manage application services with NFV in cloud environments [28].

The second stage in SFC deployment is the composition. This step defines how the VNFs will be ordered in an SFC. The second stage in SFC deployment is the composition. This step defines how the VNFs will be placed. According to Yang et al. [29], an SFC may be classified in two different types: totally ordered, in which there is a total dependency order on the VNF set, and partially ordered, which there is a dependency among a subset of VNFs, justifying the reason of the composition step. Although these functions are usually independent, their order (or part of them) must be defined carefully and is determined by the service they seek to offer. For example, considering an IoT scenario, where one can have large data to transmit, the first VNF of a SFC can perform data compression, in order to reduce the bandwidth consumption in the network [30].

The third stage is the SFC placement. It establishes where each VNF from a given chain will be hosted. The main goal of this phase is to ensure an efficient allocation, taking into account several constraints (from the infrastructure or customers) [22]. However, some authors present other additional concerns for this stage. A common approach is to divide the SFC placement into two sub steps: VNF placement and SFC chaining [31]. While placement consists in assigning VNFs to the physical infrastructure, chaining is defined as interconnecting VNFs in order to deploy a SFC. In other words, the VNF placement defines the servers where the VNFs are deployed, and the SFC chaining defines the physical links that will connect these servers. The chaining step can be done in different ways. Kiji et al. [32] considered that there are predefined paths, calculated through Dijkstra algorithm. On the other hand, Jiao et al. [33] presented a joint optimization of VNF selection and traffic steering. A placement example of the SFC considered in Fig. 1 is shown in Fig. 3. The VNFs are allocated across different distributed data centers and connected through virtual links.

The fourth step is the SFC scheduling. After composing and defining which servers the VFNs will be allocated to, the scheduling step determines when that allocation will happen. The main focus of this step is to minimize the total execution time of the SFCs allocated [22]. As highlighted by Alameddine et al. [34], the scheduling step is still a field under investigation, and received little attention by researchers. Riera et al. [35] formulated the scheduling problem as a flexible job-shop problem. Mijumbi et al. [36] proposed a Tabu-search-based algorithm to solve the scheduling problem. The work presented in [37], formulated the scheduling problem as an MILP and solved it using a genetic algorithm.

The placement step is the main focus of our systematic review. We present the main solutions used for SFC placement in distributed scenarios.

4 Methodology

The methodology used in this work to guide the systematic review is based on [38], described in Fig. 4. We used this methodology to find the works that addressed SFC placement in distributed scenarios. The methodology steps are described in the following subsections.
4.1 Activity 1: Identify the Need for the Review

As discussed previously, the use of the NFV paradigm to create virtual network functions promises to reduce the network costs while increasing the network flexibility [17]. In addition, connecting VNFs in a logical order to compose a SFC is a big challenge. Considering distributed scenarios, some aspects increase the placement complexity, such as capacity and delay constraints. Although the VNF paradigm is not a new concept and past works have presented reviews and surveys about this technology, this paradigm continues to evolve and deserves renewed attention and evaluation. Technologies, such as IoT, fog computing, CDN, e-health, stand to benefit from deploying SFCs to create customized services that meet customers requirements. In addition, technologies such as 5G brought up fundamental recent architectural changes which further raises the need for our new survey.

In the systematic review presented in this paper, we focus on SFC placement in distributed scenarios, evaluating their possible applications and goals. We identify the goals and constraints considered by the primary papers for the placement of SFCs, as well as the most common scenarios where the SFC paradigm has been applied. In addition, we identify the most common techniques applied to create placement strategies.

4.2 Activity 2: Define Research Questions

The main goal of this work is to answer the following research questions:

– RQ.1: What are the main goals considered during the SFC placement?
– RQ.2: What are the most common techniques used to define SFC placement strategies in distributed scenarios?
– **RQ.3**: In which scenarios the SFC placement has been commonly used?
– **RQ.4**: What are the most used metrics to evaluate SFC placement strategies?

### 4.3 Activity 3: Define Search String

The search string used in this work to identify relevant papers was: *("service function chain" OR SFC) AND placement.* Note that we did not include ‘distributed’ in the search string, because it would have restricted the search a great deal, returning only few articles. Despite having other nomenclatures for VNF concepts, such as the European Telecommunication Standards Institute (ETSI) Network Service (NS) [39] definition, this systematic review assumes the Internet Engineering Task Force (IETF) definitions.
4.4 Activity 4: Define Sources of Research

We considered the following databases as the main sources for our research: IEEE Xplore, Science Direct, and ACM Digital Library.

4.5 Activity 5: Define Criteria for Inclusion and Exclusion

In order to limit our scope, we considered only works published in journals and conferences in the last 10 years (between 2010 and 2020). A selected paper must focus on SFC placement in distributed scenarios. It is important to highlight that we removed works that consider SFC placement in centralized scenarios and works which do not clearly define the scenario as being distributed.

4.6 Activity 6: Identify Primary Studies

In June 2020, the search returned 117, 411, and 213 works (741 in total) from IEEE Xplore, Science Direct, and ACM Digital Library, respectively. After removing duplicate and short papers, we read all their abstracts applying the criteria for inclusion and exclusion. This activity is made manually, i.e., without any support tool. We finally selected 58 works for the ultimate data extraction and evaluation.

4.7 Activity 7: Extract Relevant Information

After reading the 58 papers selected in Activity 6, we extracted relevant information that answers the research questions previously established in the Activity 2. It is important to highlight that this activity is made without any support tool.

4.8 Activity 8: Present an Overview of the Studies

An overview of all works selected in this systematic review was elaborated in this activity, which is presented in Sect. 5, with the purpose of classifying the papers according to the research questions formulated in Activity 2.

4.9 Activity 9: Present the Results of the Research Questions

Finally, in this activity we discuss our findings and address our research questions regarding the works that present SFC placement in distributed scenarios.

1 http://ieeexplore.ieee.org/Xplore/home.jsp.
2 http://www.sciencedirect.com/.
3 http://dl.acm.org/.
4.10 Threats to Validity

In this section, we present the limitations regarding this systematic review. We have identified the following threats to validity:

- **Research question**: we formulated our search string taking into account the terminology adopted by the IETF. Despite being a widely used term, since 741 articles were raised in our searches in the search bases, some articles may not have been included in our systematic review. However, other terminologies have been considered in recent reviews, such as [21].
- **Data extraction**: as mentioned in Sect. 4.5, we considered only works that presented SFC placement solutions in distributed scenarios. The aim was to verify proposed solutions that considered relevant characteristics of these scenarios. However, other works that did not enter our research may present relevant solutions to the problem of SFC placement.

5 Results and Discussions

5.1 Descriptive Analysis

Figure 5 shows the number of articles published per year, from 2010 to 2020. The articles are very recent, mostly appearing after 2015, and up to 85% were published in the last three years: 16 in 2018, 24 in 2019, and 11 in 2020. This is a clear indication that SFC placement remains a relevant research field.

Figure 6 shows the number of works per source. One can see that most of the articles were published in the IEEE Xplore, followed by Science Direct and ACM Digital Library with 38, 20, and 2 articles, respectively.
5.2 Goals and Solutions

In this section we describe the goals considered by the papers reviewed in this systematic review. We also present the solutions proposed by these articles to achieve their goals.

Most of the articles found in this systematic review proposed mono-objective solutions, i.e., only one objective function is optimized during the SFC placement process. Figure 7 shows the main placement goals found in the primary studies. The most common goal is the minimization of operational cost (nineteen papers), followed by minimization of delay and maximization of resource utilization with ten and eight articles, respectively. Some papers also addressed other specific problems, which are not related to the previous ones, and we present them in a separated subsection. Finally, we present the papers that proposed multi-objective solutions to achieve their goals, i.e. more than one objective function is considered in the SFC placement process.
placement process. A more detailed analysis of each paper for the placement goal is presented in the following sections.

5.2.1 Operational Cost

Most of the primary studies addressed the SFC operational cost since one of the biggest advantages in adopting NFV is to achieve a flexible utilization of computational and network resources. Due to migration of network functions from dedicated hardware to general purpose servers both network operators’ operational expenditures (OPEX) and capital expenditures (CAPEX) are reduced [40]. The operational costs of a SFC can be divided in two types: costs about the resources and costs about the VNF placement [41]. The former is related to the remaining resources on servers and links from the physical infrastructure. The latter is about the costs related to the placement, such as energy consumption, license fees, among others [42][43]. Nonetheless, obtaining the optimal SFC placement cost requires careful planning. For instance, if a SFC placement strategy takes into account only the costs regarding the network bandwidth, the costs related to the node resource consumption can be high, or the placement solution can result in a high VNF migration cost [44]. In addition, minimizing the operational costs of SFCs can conflict with other metrics, such as performance and resiliency [45].

Five articles found in this systematic review addressed cost optimization for cloud computing scenarios. Chen et al. [15] considered SFC placement in cloud computing scenarios, more precisely public cloud services. The main goal was to minimize the overall cost taking as constraints QoS aspects, such as delay. The authors formulated the optimization problem as an integer linear programming (ILP) and proposed a heuristic algorithm based on Hidden Markov Model (HMM) to solve the problem for large scenarios.

Pei et al. [41] solved SFC placement problem with dynamic VNF placement for a geo-distributed cloud, considering that the placement cost includes resource cost and VNF placement cost. Unlike [15], the proposed solution focused on online SFC placement, adapting the placement strategy according to dynamic network load. Authors formulated the SFC-EP with dynamic VNF placement as a binary integer programming (BIP) model and proposed two algorithms: SFC embedding approach and VNF Dynamic Release Algorithm. The former obtains the selection, placement, and concatenation solutions of VNF instances by running the shortest path algorithm (e.g., Dijkstra) in a multi-layer graph; while the latter is responsible for the optimization of placed VNF instances while seeking to reduce their running time.

The work proposed by Feng et al [46] also addressed online SFC placement. It considered flow processing, routing decisions and at the same time the associated allocation of cloud and network resources. They focused on stabilizing the service input rate of cloud services with minimum cost. The proposed algorithms are based on Lyapunov drift-plus-penalty control methodology, which is applicable to the dynamic control of queue network [47, 48]. Tang et al. [49] also proposed a Lyapunov based algorithm for the cost optimization for SFC placement. They considered a 5G access network scenario and formulated a model for minimizing cost.
considering network stability. Afterwards, they transformed this problem into the minimization of the upper bound of drift-penalty function by applying the Lyapunov optimization technique. A genetic algorithm based heuristic was proposed for VNF scheduling and mapping.

In [50], Gupta et al. formulated a cost optimized latency aware placement (COLAP) framework for optimum SFC placement in a distributed cloud. COLAP implements a randomized selection of clouds and a heuristic for placement based on the latency prediction through support vector regression. It takes as input both tenant’s initial requests and online requests during operation with data for latency threshold and cost budgets and tariffs.

Two other articles proposed online solutions for SFC placement with concern for cost minimization but did not specifically consider cloud computing scenarios. Yao et al. [51] studied the cost minimization of dynamic SFC placement and scaling in distributed data center networks. The authors proposed an online scaling algorithm, which is composed of two parts. The first part is a Fourier-series-based forecasting approach to minimize the placement cost by avoiding frequent changes in network topology. The second part is an online deployment algorithm to properly deploy VNF instances based on online learning. An SFC placement strategy for content delivery networks was proposed by Jahromi et al. [52]. They considered that the SFC is composed of Value Added Services, that are functions for video delivery applications. The main goal was to minimize the SFC costs while QoS of all service requests were jointly satisfied, taking into account the eventual reuse and migration of already deployed VNFs from previous SFC requests. The optimization problem was formulated as an ILP and solved using the CPLEX optimization tool for a small scale scenarios. The work proposed by Hejja and Hesselbach [53] presented an offline algorithm for SFC placement with the purpose of minimizing the operational cost about the SFC migration and the energy consumption cost. A heuristic base algorithm is proposed to solve the placement in a problem in a reasonable time. It is important to highlight that the authors also proposed an online version of the algorithm, but focused on minimize the power consumption, and it will be described in the Sect. 5.2.4.

The concept of reuse of already deployed VNFs within instantiated SFCs was considered in two other articles in order to reduce the cost. Mohamad and Hassanein [54] proposed a sharing-based placement where underutilized VNFs deployed in current SFC requests should be given priority for reuse instead of instantiating a new VNF. Since the authors considered an edge computing scenario, sharing is important due to the limited computing resources of the network devices. They formulated their problem as an ILP model in order to minimize the overall deployment cost, seeking to reduce resource utilization while satisfying customers’ QoS requirements. The proposed ILP model was solved using the Gurobi solver [55]. The main objective of work proposed by Guo et al. [56] was to reduce deployment costs by sharing VNFs between different SFCs in order to avoid multiple VNF instantiation across the network. The authors formulated the SFC placement problem as a mixed ILP (MILP) and presented an algorithm for solving it. This algorithm has two main steps: an initial phase based on a centrality measure that searches servers as
candidates to place shared VNFs and an adjustment phase based on a Markov Decision Process that can adjust the shared VNFs and allocates remaining VNFs.

Two articles used reinforcement learning algorithms to optimize the cost of SFC placement: [57] and [58]. In [57], Luo et al. proposed a deep reinforcement learning framework that combines a recurrent neural network with a reinforcement learning agent to deploy in geo-distributed data centers. The recurrent neural network model was used to predict the traffic flow while the deep reinforcement learning agent was used to produce the SFC placement decisions. The main goal was to reduce the overall cost of SFC deployment in the distributed data centers. The work presented in [58] proposed a cloud-edge SFC orchestration architecture based on blockchain and deep reinforcement learning for IoT applications. Deep reinforcement learning was used to support automatic SFC orchestration and dynamic adjustment. For the placement, the SFC cost (including SFC orchestration and SFC) and constraints about the CPU, and memory resources and link utilization were considered. The use of reinforcement learning algorithms by these articles was carried out to handle the current condition of the environment, and to decide the most suitable SFC placement response: in [57] considering the pattern of traffic variation, and in [58] taking into account a high-mobility IoT network.

Similarly to [58], Nguyen et al. [59] also addressed the SFC placement problem in an NFV-based edge cloud system considering an IoT network topology. The authors proposed an optimization problem for VNF placement and chaining taking into account aggregated traffic from IoT gateways. The main goal was to find the placement solution which gives the minimum cost over the allocated components. They formulated the problem as a non-convex Integer Programming problem. Two algorithms were proposed to solve the problem: a Markov approximation approach and a node ranking-based heuristic.

Jin et al. [40] addressed SFC placement over devices located at the edge of the network. The authors proposed a SFC allocation solution for online gaming in a fog computing scenario. The resource allocation problem was modeled to optimize the total cost of SFC placement, considering CPU/GPU cost, VNF instantiation cost and transmission cost, with delay, resources and link capacity as constraints. Then, a heuristic algorithm named Probabilistic SFC Embedding based on Cost Optimization (PSECO) was proposed to solve the optimization problem. PSECO can be divided in two steps, where the first one maps the VNFs into physical network node, and the latter maps the virtual links among instances to the physical links based on the shortest path.

The work by Fang et al. [60] sought to guarantee SFC reliability and optimize the costs of network resources in the implementation of the VNFs. To this end, the article proposed two reliability protection mechanisms: an all-nodes protection mechanism and the single-node protection mechanism. For each protection mechanism, the authors formulated the problem as an ILP model and proposed a heuristic algorithm based on dynamic programming and Lagrangian relaxation.

In [61], the optimization task was to minimize the total deployment cost satisfying all the SFC requirements of the flows. SFC placement was modeled as a set cover problem and two algorithms were proposed with logarithmic approximation.
factor. The first algorithm is based on Linear programming rounding, and the second one is based on the greedy approach.

The work presented in [44] tried to minimize the cost of activating a function node (a node that can host VNF). This cost is associated with energy and maintenance costs needed to deploy the VNF in a node. A BIP model and an algorithm were provided. Note that the model is an offline version of the problem, whereas the algorithm copes with a dynamic online version.

The work [32] presented SFC placement and routing that specifies the path which passes through all required VNFs while meeting service bandwidth requirements. The main goal was to minimize total cost associated with VNF placement and link usage. The authors then proposed an ILP problem formulation, considering reduction of the number of VNF instances, reduction of link usage and cost minimization. Then, a heuristic was proposed to handle cases where the ILP is intractable.

Tastevin et al. [62] considered a problem of allocating SFCs over an infrastructure composed of commodity servers, called points of presence (PoPs), with VNF hosting capabilities. PoP can be defined as an access point in an IP network owned by a telecommunication carrier [63]. A PoP usually is composed of interconnected co-located backbone routers but may also represent other equipment, such as servers and switches [64]. They proposed a cost-driven ILP formulation for this problem. To solve it, a graph-based heuristic that combines graph centrality and multi-stage graphs was proposed.

Tashtarian et al. [65] put forward a solution for SFC placement with different strategies: NFV parallelism, NFV distribution, optimal resource allocation, and optimal data rate. NFV parallelism and distribution allow to execute multiple replicas of a single VNF and distribute the traffic among them. The authors defined only one function that jointly covers four cost aspects: the total cost of deployment, energy costs, distribution costs, and parallelism costs. This function is considered in the placement problem, which was modeled as an MILP and solved using the greedy algorithm.

Evaluating the works that addressed the optimization of the SFC operational cost as the main goal, all of them considered the allocation cost of nodes (server, switches, etc.) and links. The costs about the nodes are usually related to the cost of allocating a VNF into the nodes according to the available computational resources (CPU, memory, and storage), where more powerful nodes increase the placement cost. The cost of the links are usually related to the network bandwidth. However, the work presented in [61] does not provide details about the operational cost. The authors only define the cost of placing a VNF into a physical node.

In addition to the costs about the nodes and links, some works considered other aspects of SFC operational costs. The work presented in [40], which is focused on SFC for online mobile games, considered the costs of CPU and GPU allocated to run the VNFs. In [46, 51] the authors considered the processing cost, which is related to the additional cost of deploy/execute a VNF in a node. Some works are more specific and define the VNF deployment cost as a “renting cost” [44, 49, 57]. Luo et al., for instance, defined the VNF deployment cost as “for renting a virtual machine or container with required resource configuration for running the instance” [57]. Liu et al. [44] and Tang et al. [49] also considered the renting cost of deploying...
a VNF in a server, but such cost is inversely proportional to the amount of available resources in the server. The works presented in [57] and [65] considered the operational cost of migrating the VNF’s image to another server. In [57], the authors considered the cost of copying the VNF’s image to the data center and launching a VM/container with the image, while in [65] the placement cost is proportional to the cost of transferring the image and booting that VNF on a physical machine.

Regarding the operational cost of links, the works presented in [15] and [40] considered more detailed aspects of cost related to bandwidth. In [15], a multi cloud scenario is considered and the authors considered that links inter and intra clouds have different costs. In [40], where a mobile scenario is considered, the cost about the transmission also takes into account the wireless transmission cost between users and small base stations, and wired transmission cost among the small base stations.

Finally, two works also considered the costs of SFC migration. In [52], the migration cost is defined as the total costs for migrating the already-deployed VNFs from one server to another. In [58], SFC migration cost is composed of the data transfer rate, the link resource usage and the data transfer delay. After the migration, the authors also consider the cost of network configuration that takes into account the network reconfiguration overhead on a server.

5.2.2 Delay

The network delay is a metric that deserves attention in SFC placement, specially in distributed environments [65]. In the SFC context, the delay is the time that the flow takes to get from the source to the destination, passing through all the VNFs from an SFC, considering the link and processing delay [66][67]. Therefore, the overall delay can be considered as the sum of the processing delay by each VNF that composes the SFC and the communication delay of links. The VNF processing delay, which is higher than the dedicated hardware due to the virtualization, may vary depending on the virtualization type (traditional virtual machines or containers) and the hardware used [68]. The communication delay is associated with the different delays when the traffic is routed through, for example, different data centers, since different links may have different associated delays and resources [57].

A typical example of VNF distribution is that of a cloud computing scenario, where the VNFs of a SFC are deployed in different racks or even in different geographically distributed data centers. This geographical distribution combined with unpredictable delay variations due to sharing of physical infrastructure among different services, can result in violation of delay requirements, resulting in penalty costs [69]. Therefore, minimizing SFC delay is an important concern for network operators.

All works presented in this subsection considered the SFC delay as the optimization goal. However, other links aspects are also considered as constraints, such as the bandwidth [70–73].

Five articles have considered SFC placement problem in geographically distributed cloud or multi domains while taking delay as their main constraint. The optimization of end-users delay with optimal placement of SFCs was addressed by
Bhamare et al. [74]. The multi-cloud scenario considers some constraints like total deployment cost and SLA requirements. An ILP formulation was proposed and a polynomial heuristic “affinity-based allocation” was presented to cope with the complexity of ILP in larger scenarios. Bhamare et al. also proposed a solution in [75] for the placement of SFCs in multi-cloud environment for CRANs, where the goal was to minimize the overall response time in a multi-cloud RAN. The placement problem was modeled as a combinatorial optimization problem and two different approaches were proposed for large networks: branch-and-bound and simulated annealing. In addition, some enhancements to the standard branch-and-bound algorithm were presented in the solution proposed by the authors.

Martín-Pérez et al. [71] solved the SFC placement problem in a multi domain scenario, considering federated clouds. They proposed six algorithms to minimize the end-to-end mapping delay. All algorithms are variations of a greedy approach and search the closest servers to deploy the VNFs meeting computing constraints. Depth first search and breadth first search algorithms were used to boost up the mapping performance in terms of running time, while using Tabu search was used to minimize the delay. In [70], Xu et al. addressed the low latency multi-domain security service chain embedding problem, but for mobile edge computing in 5G networks. The problem of cross-domain service chain embedding is based on two sub-processes: service chain partition and service subchain mapping. The authors formulated this problem as two ILP problems in order to obtain an optimal solution in small networks. Then, a heuristic solution was proposed to improve the classical Viterbi algorithm [76] to obtain a near-optimal solution for large networks.

Still considering 5G networks, in [73], Subramanya et al. proposed a machine learning based solution to predict the number of VNF instances needed to meet the traffic demand in 5G networks. Similar to [70], an ILP formulation was proposed for the latency-optimal SFC placement problem, but the adopted solution was different: a neural network was used to predict the number of VNFs taking as input measurements from the network.

Sun et al. [77] addressed the SFC placement in edge networks, similarly to [70]. The authors proposed a solution to solve the SFC request orchestration considering a SFC request as a workflow based service request while seeking to reduce the overall delay in edge computing. Thus, they sought to find a network service placement schema with minimal delay. Two algorithms were presented to solve the problem: a heuristic to map workflow-like request minimizing response time; and another algorithm, similar to the previous one, but it prohibits the VNF in the same network layer from being deployed on the same substrate node.

Two articles addressed the SFC delay optimization taking into account fog computing networks. The work proposed in [78] presented a controller to allocate container-based SFCs in fog computing environments and focused on a prototype solution. The controller was implemented as an extension of the Kubernetes platform. The default container scheduler of Kubernetes calls the extension proposed when a scheduling decision is needed in order to minimize the “latency established by the

---

4 https://kubernetes.io/.
calculation of the shortest paths or based on the target location for the pod deployment" [78]. If the latency-aware schema is selected, the SFC controller best candidate selects a node based on the calculation of Dijkstra’s shortest path algorithm. When the location-aware scheme is selected, the selected node is chosen based on latency minimization depending on the target location defined by the network manager.

Zamani et al. [72] proposed a solution for SFC placement considering the fog-to-cloud scenario for use in IoT networks. The main objective was to minimize the overall latency taking bandwidth consumption of IoT devices as constraints. The SFC placement was formulated as an ILP problem but the authors failed to state which algorithm they would use to solve the formulated model. According to their approach, the functions can be deployed in either fog or cloud nodes. The SFC mapping was divided into two steps: firstly, the possible configurations to deploy the functions are selected satisfying various constraints on nodes and links; secondly the route from the source to destination is defined satisfying various links’ constraints.

Ren et al. [30] also considered the SFC placement problem in an IoT scenario and the main goal was to process IoT data as early as possible, then, the VFN should be deployed in nodes as close to the data source as possible, reducing the overall network delay. Therefore, the authors formulated an MILP problem to minimize the actual network node chain (ANNC), which is the distance from the VNFs of a SFC to its data source, taking into account the service requirements and resources limitations. A composition mechanism was proposed for SFCs placement in IoT scenarios, and a variation of a genetic algorithm was used to minimize the ANNC.

Cai et al. in [67] proposed an algorithm for parallel placement of SFCs in distributed networks. The objective was to minimize the average SFC delay over all of the parallelized SFCs taking into account servers and link constraints. An algorithm was proposed to convert a serial SFC deployment into a parallel one, since some VNFs could run in parallel. Next, another algorithm was proposed to allocate the VNFs (both parallelized and serial VNFs) into servers, considering the total delay, VNF resource and bandwidth requirements.

Considering the works that addressed the delay minimization, the common approach is to consider the delay about the links chosen to place the SFCs, i.e., to forward the traffic among the VNFs. However, there are different delays that can be considered in the end-to-end communication of the SFC. For instance, as it was considered in [74, 77], and [75], the overall SFC delay is divided into two categories: transmission delays associated with links and computational delays associated with the clouds. The transmission delay of the scenarios considered in the works can be very high and potentially have a big impact on the end-to-end SFC delay: [74] considered a multi-cloud scenario, [77] considered an IoT scenario, and [75] considered a CRAN architecture for mobile networks. The transmission delay in centralized scenarios is too slow and has a smaller impact than distributed scenarios. The consideration of the computational delay is also important because the virtualization can increase the VNF delay in comparison to
the dedicated hardware. However, the VNF processing delay is still neglected by most of the works that tried to minimize the SFC delay.

Similar to [74] and [75], the work presented in [73] also considered the processing time as required for all VNFs in an SFC to apply a specific network operation on the arriving packets. However, they considered several delay types in the scenario, such as the user equipment processing delay, over-the-air transmission delay, hybrid automatic repeat request (HARQ) retransmission delay, and the transmission delay between the base stations and the MEC layers. Therefore, it is important to consider the particular delays that can emerge depending on the scenario evaluated, since they can impact the overall SFC delay.

5.2.3 Resource Utilization

Optimal resource utilization is another concern for network operators when making SFC placement decisions. The optimal resource utilization can be translated into minimum cost and maximum profit by efficiently utilizing computing and network resources such as CPU, memory, and link bandwidth, at the same time improving the network performance, availability and reliability [79].

During the course of this systematic review, we discovered two articles that addressed resource utilization in edge computing scenarios. Yang et al. [29] studied how to place SFCs in edge and public cloud environments considering a delay-aware VNF placement and routing problem. The main goal was to minimize the maximum link load and at the same time satisfying user’s requested delay. The problem was formulated as an integer nonlinear programming and transformed in an ILP. The authors developed a polynomial-time approximation algorithm called Randomized Rounding VNF placement and routing algorithm to solve this problem. Li et al. [80] also considered the SFC placement in edge computing networks taking into account the link resource usage minimization. In the context of their work, a VNF can be placed across the network through duplicated copies in order to reduce bandwidth consumption. An ILP formulation was developed. It considered the mapping of a series of SFC requests to data centers and edge infrastructures. Next, a priority based greedy heuristic was proposed to map SFC requests to data centers with an adjustment process to merge the VNF requests demanding for the same type of VNF.

Two articles found in this systematic review considered the resource utilization minimization in PoP. The model proposed by Luzielli et al. [68] aimed to minimize the number of virtual network functions instances mapped on a PoP infrastructure. They considered three phases: placement, assignment, and chaining. Then, an ILP model was proposed considering virtual network functions to be placed in PoPs and SFC requests. To solve the modeled problem, the authors developed a heuristic that performs a binary search to adjust the number of network functions to be instantiated dynamically (reducing the resource consumption). The work presented in [81] also considered PoP infrastructures, but focused on SFC availability. The authors highlighted that the link congestion and failures can impact SFC availability. Therefore, they proposed an optimization model to minimize maximum load on links while deploying SFCs. Khebbache et al. [82] did not address SFC placement in PoP infrastructures specifically, but they considered the SFC placement in large scale
network infrastructures. It is a relevant problem due to the complexity to find good solutions in an acceptable time. Thus, the authors presented a VNF-forwarding graph chain placement problem that finds subsets of cycles in the physical graph that can host the SFCs and meet their flow requirements. Due to the nature of the problem, the authors associated it to the well known perfect 2-matching problem which consists in finding cycles in the substrate graph with a minimum amount of resources, verifying the resource limitations and constraints.

We found two articles that addressed resource minimization in SFC placement considering telecom infrastructures. In [83], Carpio et al. proposed a solution for SFC placement in an Evolved Packet Core (EPC) network, which is the core of 4G networks. The main goal was to minimize the link and server utilization considering load balancing with constraints on the available resources in mobile core networks. An ILP formulation was proposed by the paper but no algorithm was suggested to solve it. The work proposed by Li et al. [84] presented a solution to optimize either CPU usage or bandwidth resource utilization in 5G networks. They also proposed an ILP formulation for the optimization problem, but used reinforcement learning to solve it. The ILP formulation can be adapted according to the objective of the network operator: either to minimize bandwidth consumption or to minimize the CPU consumption. In order to mitigate the ILP complexity, an adaptive deep Q-learning based SFC mapping approach was proposed. The reinforcement learning agent learns how to make decisions based on two low-complexity heuristic SFC mapping algorithms.

Similar to the work presented in [84], Song et al. [85] presented a solution to minimize the number of CPU cores and the communication resources, in terms of link bandwidth; but in generic distributed networks. An ILP formulation was proposed to minimize the objective function considering several constraints in terms of computing resources capacity, link capacity, and order of VNFs within a SFC. A heuristic algorithm based on Hidden Markov Model was proposed for SFC placement.

Most works that addressed the resources utilization optimization focused on links resources, but with different concerning. The work presented by Shang et al. [81] focused on to minimize the maximum link load, while other focused on to minimize the bandwidth consumption [80, 82–85]. Although the objective is similar (minimizing the resource consumption of the links), these different approaches result in different mathematical formulations. The work proposed by Yang et al. [29] presents a different approach to minimize the resource consumption of links: they considered the link load ratio. The main goal is to avoid situations where one or more links are highly loaded, while many links are less loaded.

Besides the concerning about the links resources consumption, the server resources consumption minimization is also considered for some works. For instance, the main focus of the work present by Li et al. [80] was to mitigate the CPU consumption and the memory consumption of the servers where the VNFs are placed. Li et al. [84] tried to minimize the CPU remaining rate of the servers. These works also minimized the bandwidth consumption as mentioned previously. Different of previous works, the works presented in [68] and [83] tried to minimize the computational resources in a different way. In [83], Carpio et al. tried to minimize the number of nodes where the VNFs will be placed, in order to reduce the number...
of nodes with computing resources consumed by the VNFs. In [68], a VNF replica can be placed in more than one server, consuming more resources from the server, then the goal is to minimize the number of replicas allocated in the infrastructure (reducing the computational resources consumption).

5.2.4 Energy Consumption

Energy consumption reduction is becoming increasingly important due to the explosive growth of data center traffic and power consumption that result in increasing costs and carbon footprint [86]. Some studies indicate that the adoption of the NFV paradigm can save up to 50% in energy consumption as opposed to traditional network infrastructures [87]. However, to achieve a considerable energy consumption reduction, the placement algorithms need to consider this relevant metric in the SFC placement decisions [86].

The reduction of power consumption in multi-domain scenarios was addressed in two articles. In [31], Kouah et al. formulated an offline optimization problem as a Mixed Integer Program (MIP) to minimize the SFC energy consumption in IoT scenarios. The authors considered the SFC placement not only for IoT equipment as well as in conjunction of a cloud infrastructure. The problem was solved using a common solver (Gurobi [55]). A Genetic Algorithm for reducing solution time was also proposed. The work presented in [88] also addressed the energy consumption optimization for SFC placement in multiple domains, more specifically, it considered a multiple cloud scenario. Different from [31], the Sun et al. [88] formulated the problem as an ILP and a low-complexity heuristic algorithm for SFC orchestration across multiple domains sufficient for obtaining a near-optimal solution.

In [89] and [90], the authors focused on energy consumption optimization of SFC placement across telecommunication networks. An ILP formulation was proposed to minimize the SFC energy consumption while considering as constraints the links and servers capacity. A Markov-based algorithm was adopted to find the near-optimal solution in a polynomial time.

The work proposed by Hejja and Hesselbach [53] presented an online algorithm for SFC placement with the purpose of minimizing energy consumption. They proposed an online algorithm that places the arriving SFC. To minimize the energy consumption, the algorithm proposed monitors the servers in the infrastructure, and after remove an SFC (due to expired lifetime) it turns off all servers that do not host a VNF and do not receive traffic to process.

Evaluating the works that proposed solutions to optimize the energy consumption, one can note that most of them considered the energy consumption about the computing and communication. For instance, Sun et al. [88] considered that the computing energy consumption is generated by the servers where the VNFs are hosted. A server consumes energy if it is on, even without VNFs placed. If a server hosts VNFs, its energy consumption increases according to the workload. On the other hand, the forwarding energy consumption is related to the energy consumed by a server to forward data to another servers. The forwarding consumption depends on the number of ports used in the communication with other servers. The works presented by Xu et al. [89] and Zhang et al. [90] defined that the energy consumption
of SFCs in a Telecom network scenario is composed of two main parts: energy consumption of servers hosting VNF instances and the energy consumption of physical links transmitting traffic. The server energy consumption consists of two parts: the energy consumption to keep the server on, and the energy consumption produced to process the VNF demands of SFC requests, which is related to the CPU utilization. The energy consumption about the links “depends on its on/off state and bandwidth utilization [89].”

However, two other works considered different energy consumption models. The work presented by Hejja and Hesselbach [53] considered that the energy consumption only about the servers that is composed of the sum server’s idle power and the power consumption due to the traffic load, i.e., the utilization of that server. Kouah et al. [31] considered an IoT scenario for the SFC placement, and the energy consumption can be defined as a sum of three energy consumption of three different aspects: communication, processing, and sensing. The communication energy consumption takes into account the energy consumption for transmit bits between two IoT nodes, which varies depending on the distance between these nodes and the communication technology used (WiFi, Bluetooth, Zigbee, Sigfox, LTE, and so on). The processing consumption is about the energy consumed to processing a flow in a specific node, and depends on the amount of the computational resource required by the VNF. Finally, the sensing energy consumption is the consumed energy when the sensing function of the IoT device is activated. The sensing function is activated according to the application demand.

### 5.2.5 Revenue

Maximizing revenue is a common objective of network operators, but is hardly a trivial task. The revenue maximization is not an exclusive goal of network managers, which provides SFC services to their customers. Several works tried to optimize the revenue in cloud environments, which also is based on virtualization, i.e., launching virtual machines to host applications [91–94]. However, the way in which revenue will be maximized may differ in an SFC scenario. One possibility may be to find the right trade-off between request acceptance and rejection rate of SFC requests [95].

A big challenge in SFC placement is to find the trade-off between admitting as many requests as possible while reducing the operational costs. For example, data centers in different locations have different energy consumption costs that impact the network operator revenue [96], thus choosing the most adequate data center for a SFC can dramatically impact revenue.

Two articles found in this systematic review addressed the revenue optimization in SFC placement: [97] and [98]. Both articles formulated the optimization problem as an ILP, but adopted different algorithms reflecting their different scenarios. Xie et al. [97] studied the problem of placing SFCs in a dynamic scenario, i.e., one with time-variant service flow sizes and VNF sequencing. The main goal was to maximize the revenue of the telecommunication service providers. Two algorithms were proposed to solve the optimization problem: the first one called DynAmic vnF placementT (DAFT), was designed to solve the formulated problem by adopting a primal–dual technique. The second one called Feasible DAFT (FDAFT) complies to
capacity constraints when creating an auxiliary graph for solving the general problem. Li et al. [98] considered the SFC placement across multiple vendors, which tend to be very different due to vendor dissimilarities. Due to the variability of scenarios, the authors proposed a reinforcement learning-based algorithm for the revenue maximization taking into account service benefit, resource cost, and service quality of candidate SFC paths.

Evaluating these two works, one can see that the revenue is calculated based on the gains of meet an SFC request and possible penalties for not meet the SFC request or for not adequately meeting. For Xie et al. [97], the total revenue is calculated based on the revenue of serving an SFC request and the penalty of not serving an SFC request. The goal is to maximize the total revenue of a Telecom service provider considering several SFC requests over the time. On the other hand, for Li et al. [98] the revenue is calculated based on the service benefit, the service cost, and the service quality of the deployed SFC. These parameters are calculated based on the path (set of links) selected to place the SFC.

5.2.6 Other Problems

In this subsection, we present articles that addressed different problems from those illustrated in the previous subsections. We identified three articles that fell in this category.

The work presented in [99] proposed a SFC placement solution for multi-domain scenarios. The proposed solution offers a heuristic based algorithm for SFC placement in multi-domain scenarios taking into account context information. This algorithm aims to mitigate the bandwidth over-provisioning problem and to optimize the revenue-cost ratio (from CPUs and bandwidth). A greedy based algorithm is proposed to allocate the SFC, considering the computational resources and link constraints.

In [100], a solution to maximize the acceptance rate of SFC requests was proposed considering as constraints the satisfaction of subscribers preferences. An ILP formulation for the optimization problem was developed by the authors as well as an algorithm to solve it in an offline and online modes.

The problem addressed in [101] considered users roaming over a cellular network (from cell to cell) and a SFC (for an undefined service) attending each respective user. The VNFs of these SFCs are allocated on MECs directly linked to the cellular network and linked between them. The main goal was to minimize the service interruption (downtime) for users. The SFC placement problem was formulated as an ILP and a heuristic was proposed to solve it. The heuristic can be divided into two parts: placement and migration. In the placement phase, the SFCs were deployed in servers considering delay constraints. In SFC migration, upon a user’s handoff, the VNFs of a SFC were migrated among the servers in a way that avoids service interruption.
5.2.7 Multi-objective

Other articles found in this systematic review proposed problem formulations with two or more optimization objectives. These multi-objective optimization problems are more complex than the mono-objective ones because in most of the cases there conflicts among such objectives and a balance must be drawn [102].

Bouten et al. [103] proposed a solution for SFC placement considering affinity and anti-affinity constraints. Since the service providers are free to define their SFC constraints, it is possible that conflicting constraints are introduced. Then, the SFC requests are evaluated using a semantic approach in order to check their validity. Multiple optimization objectives are presented: maximizing revenue and the acceptance ratio; minimizing the number of substrate nodes; and minimizing the total bandwidth consumption. A heuristic based approach was proposed to solve the optimization problem.

Gupta et al. in [104] considered a problem to dynamically provide placement of the complete SFCs belonging to a virtual network service in multi-cloud systems. They proposed the P-ART framework which is an innovative predictive dynamic placement algorithm that considers changes in the state of the cloud environment to ensure placement validity. The framework takes into account cost and delay requirements from customers for the SFC placement.

A multi-domain network SFC placement problem was addressed in [105]. A framework was proposed to jointly optimize the energy consumption of servers and balance the load. Therefore, a heuristic algorithm was adopted to solve the multi-objective optimization problem in a feasible time.

In [106], Troia et al. addressed the dynamic provision of slicing physical 5G networks into logical networks in order to provide customized data transport to each service. Then, ILP models were proposed to minimize “the probability of blocking traffic requests, energy consumption of physical network devices and interruption of service due to the reconfiguration of the slices”.

Harutyunyan et al. [107] also proposed a 5G slicing mechanism, that represents each slice as VFNs which compose a SFC. The VNFs of a SFC can be allocated in hierarchical data centers with a different number of computing resources. The main objective was to minimize the bandwidth consumption in the transport network, the service provisioning cost, and the number of VNF migrations. The placement problem was formulated as an MILP and a heuristic was proposed to address the scalability problem. In [108], a SFC placement solution for 5G networks was also proposed. It mainly considered minimizing the delay of request services, service provisioning cost, and the VFN migrations. Three different ILP formulations were presented to optimize each objective function. They also provided an end-to-end delay model suitable for SFCs for 5G mobile networks. In order to avoid the scalability issue of ILP formulation, a heuristic was used.

Jin et al. [109] also considered cellular networks, but the SFC was a general signaling processing flow of virtual Mobility Management Entity (vMME) from LTE (4G). Several cost functions were minimized in the proposed problem: the total signaling communication overhead cost; signaling communication overhead cost on backhauls; and migration overhead cost. Since the placement problem is NP-hard,
the authors proposed three different heuristic approaches based on genetic algorithms to solve the problem.

The work presented in [110] proposed a solution to jointly optimize positioning and routing of SFCs to minimize the operational cost and network congestion. The authors modeled the optimization problem as an ILP one and proposed an approximation algorithm called Candidate Path Selection (CPS). CPS solves the offline problem with a bounded approximation ratio. The authors also proposed an online algorithm (OCPS) that deals with the lack of future information and the rapidly fluctuating demand.

Eramo et al [111] proposed a solution to minimize operation costs (revenue loss and energy consumption). This problem was broken down into two others: the first one allocates a set of SFCs seeking to maximize the amount of accepted bandwidth considering peak hour bandwidth demand, i.e., SFCs demanding more bandwidth are prioritized over SFCs demanding less bandwidth; and the second one tries to consolidate VNFs on servers to minimize energy consumption costs during the hours when traffic demand is lower, taking into account also the revenue loss due SFC reconfiguration (migration imposes QoS degradation to SFCs). Three heuristics were proposed: one for the first problem and two for the second one. The first heuristic follows a greedy approach based on link and node stress. The second one greedily migrates VNFs from servers with low power consumption and high bandwidth demand to servers with high power consumption and low bandwidth usage. Finally, the third algorithm uses a Discrete Time Markov Decision Process to determine the policy to be applied at each hour interval to reduce the total operational cost.

The work presented in [112] proposed a Tabu search scheme to deploy SFC in fog devices taking into account energy and resource constraints in a fog/cloud architecture. The main objective was to minimize the SFC delay and to improve load balancing (distribution) across network nodes. The authors proposed a solution based on both a heuristic and a metaheuristic. The Tabu search was used to define the SFC placement, while graph-based shortest-path heuristics were used to define the connections of nodes where the VNFs were deployed.

Tajiki et al. [113] addressed the problem of SFC placement in fog networks considering the trade-off between reliability, QoS, and energy consumption. To solve the corresponding problem, they mathematically formulated an optimization problem called the Optimal Fog-Supported Energy-Aware SFC rerouting algorithm (OFES) and proposed a near-optimal heuristic called Heuristic OFES (HFES).

5.3 Scenarios

The primary studies selected in this systematic review considered different application scenarios for the SFC placement, and mainly concentrated on networks of Internet Service Providers (ISP), 5G environments, smart city, Mobile Edge Networks (MEC), fog computing, hybrid networks, and Content Delivery Networks (CDN).
Some works do not consider a specific scenario or application to evaluate the proposed solution. The works consider either randomly generated topologies, topologies defined in simulators, or prototypes, but without specifying a specific scenario. We classify these works as “other”. Figure 8 shows the number of articles for the different scenarios addressed.

Most of the papers (35 out of 60) considered different network topologies based on the different worldwide ISPs [15, 29, 29, 32, 41, 44, 46, 50, 51, 54, 56, 57, 60–62, 67, 68, 74, 77, 80–85, 89, 90, 97, 99, 100, 103–105, 110, 113]. In this context, we can highlight applications considering the network topologies of China’s Telecom [67], USNET [60, 85, 89, 90], NSFNET [29, 32, 54, 60, 85, 89, 90, 97], The German backbone network [61, 100], Pan-European [100], Chinese CERNET2 [105], the ATT north America network [84], and Google data centers [57], which also acts as an ISP.5 have also been the scene for SFCs. However, the ISP network can be generated through simulation software. For instance, in [74], bhamare et al. used the GUESS software to generate the network topology used in the experiments. The topology consists of user clusters, where a user cluster is an ISP network with a 3-tier hierarchy of routers: access, aggregation, and core routers. The choice for this type of scenario reflects the authors’ need to evaluate solutions considering a large numbers of geographically distributed nodes.

The second most popular scenario, present in seven papers (or 11.6%), was 5G networks [49, 53, 71, 73, 75, 88, 107]. These articles were motivated by the need to meet two main 5G networks’ requirements: low energy consumption and low latency.

The smart city scenario was considered in [31, 78, 109], and [106]. This type of scenario has interesting dynamic characteristics due to the high mobility of users, which were presented as one of the major challenges in the SFC placement [109]. Specifically, this work [109] considered the SFC placement to meet the scenarios of vehicular networks and shared bicycles. Besides, authors seek to meet other scenarios, such as smart home and smart metering. A surveillance camera solution was

5 https://fiber.google.com/.
presented in [78], considering face recognition services distributed over different nodes. The studies in [31] and [106] chose the application of solutions based on generic topologies, with nodes representing IoT gateways.

We also identified four papers that proposed solutions applied to Mobile Edge Computing (MEC) scenarios [58, 70, 101, 108]. One of the main characteristics of scenarios based on the MEC’s concept is the ability to bring computing and storage resources close to the edge of the network, to the end-users, by pairing small servers with base stations. Generally, these servers have limited computing power and storage space, requiring that the service function allocation process be optimized to meet demand and fulfill such restrictions. In this sense, [58] addressed the process of minimizing the resources used in this scenario to meet a set of restrictions. In [70, 101, 108], authors considered reducing latency in the SFC placement process.

We also identified four papers that proposed solutions applied to Mobile Edge Computing (MEC) scenarios [58, 70, 101, 108]. One of the main characteristics of scenarios based on the MEC’s concept is the ability to bring computing and storage resources close to the edge of the network, to the end-users, by pairing small servers with base stations. Generally, these servers have limited computing power and storage space, requiring that the service function allocation process be optimized to meet demand and fulfill such restrictions. In this sense, [58] addressed the process of minimizing the resources used in this scenario to meet a set of restrictions. In [70, 101, 108], authors considered reducing latency in the SFC placement process.

Three papers considered scenarios based on fog computing [30, 40, 59, 72, 112]. Unlike MEC-based scenarios in which processing is undertaken by the edge server that receives requests, the concept of fog computing allows data to be processed within a fog node or an IoT gateway placed within the user’s LAN. Such node provides higher computing power and acts as a centralized unit at the edge of the network, serving a set of small distributed servers. In this scenario, [30, 40] have as their main motivation to optimize the use of resources in the face of Fog’s limited computing power. In [112], the objective was to minimize delays in the process of SFC placement. The paper presented by Zamani and Sharifian [72] considered the SFC placement in a fog-to-cloud scenario with the purpose to minimize the overall latency and bandwidth consumption. In Nguyen et al. [59], a SFC placement was presented in an IoT edge-to-cloud scenario to minimize the operational cost of the scenario.

In [52], the proposed solution considers SFC placement in a Content Delivery Networks (CDN) application. A characteristic of this scenario is the delivery of content considering cache servers at the edge of the network with different content update policies. One of the main challenges in this scenario was to minimize the SFC costs while the QoS of all service requests are jointly satisfied, given an exponential increase in content consumption. Hence, the solution presented in [52] aimed to minimize operating costs in this type of scenario.

Four other articles considered generic scenarios for evaluating their proposed solutions [65, 98, 111, 114]. Although they did not present details of the scenarios considered in the articles, they all consider that the nodes were geographically distributed. Tashtarian et al. [65] considered a network topology composed of six physical machines and nine links (a figure is presented in the paper to illustrate the topology). However, no more details about the topology generation were provided. In [111], the network topology is composed of five core nodes, five edge nodes, and six access nodes. No more details about the network topology generation were provided. The CloudSimSDN was used in [114] for the generation of the network topology used in the evaluation. The authors consider a network composed of multiple distributed cloud data centers, but do not follow a specific scenario or real topology. Different from previous works, Li et al. [98] considered both simulated and prototype topologies in their evaluation. The simulated topology is composed of six
| Table 1 | Metrics used in the evaluation of solutions proposed by recent articles |
|---------|---------------------------------------------------------------|
|         | Metrics | Reference | Description |
| Energy  | Power consumption | [53, 106, 112, 31, 88, 105, 113, 89, 90, 111] | The energy consumption of the proposed solution was used as an evaluation metric |
| QoS     | Delay    | [30, 59, 58, 78, 50, 72, 112, 73, 77, 15, 100, 104, 74, 75, 41, 46, 29, 70] | Delay, bandwidth, and network congestion were metrics used by the studies to assess the quality of services, defining them as constraints in evaluating the proposed solutions |
|         | Bandwidth | [30, 67, 78, 80, 29, 89, 90] | |
|         | Congestion | [110] | |
| Cost    | Operational cost | [49, 59, 44, 56, 32, 82, 50, 60, 15, 104, 115, 53, 101, 75, 105, 51, 99, 52, 62, 85, 109–111] | Costs involving some solutions are also considered. Here, the operating cost considers the cost of infrastructure equipment necessary to meet demand and budget constraints and maximizing expected revenue |
|         | Revenue | [82, 98, 99] | |
| Resource utilization | Resource utilization | [65, 68, 59, 81, 67, 103, 40, 58, 73, 80, 15, 54, 29, 75, 51, 84, 83, 107, 111, 113, 108] | The use of available resources was one of the most evaluated metrics among the studies found. Here the use of CPU, memory, and link are observed as restrictions to achieve the defined objectives, in addition to the number of nodes distributed geographically, and the number of VNFs instantiated to meet the demand |
|         | Number of nodes or VNF instances | [80, 29, 89, 90, 40, 56, 41, 97] | |
| Load balancing | Load balancing | [97, 103, 105] | Load balancing restrictions were also presented as evaluation metrics in articles found in the literature |
| Reliability | Availability | [31, 81, 113, 115] | The studies also considered metrics that measure the reliability of systems. The constraints defined by the algorithms were defined based on the availability, acceptance rate, and the services’ utilization rate |
|         | Acceptation rate | [97, 44, 56, 98, 67, 82, 50, 60, 54, 104, 29, 100, 71, 101, 99, 107, 29, 41, 70, 108, 114] | |
|         | Utilization rate | [97, 41] | |
| Scalability | Throughput | [44, 41, 46] | Some studies presented the evaluation of throughput as a metric to measure the scalability of the proposed solution |
| Algorithm | Execution time | [65, 68, 32, 56, 78, 103, 50, 60, 31, 73, 100, 115, 51, 84, 62, 107, 41, 46, 29, 70, 111, 108] | The average execution times and convergence times of the algorithms in the SFC placement process were used as metrics to evaluate the proposed solutions |
|         | Convergence time | [59, 82] | |
nodes (not following a specific topology), while the prototype is composed of five connected containers, that represent the servers in the network.

5.4 Metrics

The articles found in this systematic review evaluated their results based on different metrics to achieve the defined objectives, such as minimizing energy consumption, maximizing throughput, among others. Table 1 shows the metrics used by each primary study of our systematic review.

We observed that energy consumption was considered as a restriction in ten studies. Another twenty-six studies considered the quality of service metrics to evaluate the solutions. Among these, eighteen studies considered delay, seven considered available bandwidth, and only one considered network congestion as evaluation metric.

One of the most popular metrics among the articles was the solution’s operational cost, being considered by twenty-five out of sixty papers found. Another three articles considered revenue as an evaluation metric.

Metrics related to the use of resources were also considered, totaling twenty-one articles. Here the use of CPU, memory, and network links were observed as constraints to achieve the defined objectives. In the same vein, eight studies analyzed the number of nodes used by the proposed solution and the number of instances of service functions necessary to meet the demand. Another three studies analyzed load balancing.

The reliability of the services to be allocated was also taken into account by twenty-seven studies, in which four of them considered availability, twenty-one considered acceptance rate, and two the utilization rate. Another three articles presented the evaluation of throughput as a metric to measure the proposed solution’s scalability.

Finally, twenty-four articles evaluated the average execution and convergence times of the algorithms presented in the SFC placement process.

6 Discussion

Although the NFV paradigm was proposed by ETSI as early as 2012 [116], one can see that it is still a relevant research topic, mainly considering the management of SFCs. Most works found in this systematic review are very recent, being published mostly in 2020, 2019, and 2018. The high number of recently published papers only reinforces the importance of the NFV paradigm in the development of new network services and architectures. Indeed, new network based applications arose in the last years, which demand specific network functions such as IoT, vehicular networks, e-health, smart city applications, among others. In addition, with the growth of smartphone popularity, the softwarization of the cellular networks, more precisely in the next generation (5G), is a trend to develop new network architectures and functionalities for mobile systems [117].
Observing the scenarios considered in the works analyzed in this systematic review, traditional ISP networks were the most frequent distributed scenario evaluated in the literature. It is justifiable because ISP infrastructures are suitable for software-based solutions, since ISP providers are already familiar with the cloud computing paradigm [118]. Currently, the Internet Service Provider Partners (ISPP) infrastructure is already composed of distributed data centers supported by SDN switches and commodity servers [119].

We also found works that presented SFC placement solutions for emergent technologies such as IoT, fog computing, smart city, CDN, and MEC. In addition to being highly distributed, some of these scenarios are composed of devices with limited computational resources that hinder the SFC placement, such as fog computing, MEC, and IoT. These scenarios have strict requirements which must be ensured by the network operator. For instance, IoT devices have energy constraints and limited computational resources [113] [40], 5G networks enforce requirements about latency, users mobility, connection density, peak data rate, etc [120]. Therefore, the SFC placement solutions must consider all these constraints and specificity scenarios in order to propose strategies to efficiently orchestrate SFCs.

Regarding the solutions, most works presented a mathematical formulation based on ILP (or other mathematical programming formulations) for solving the SFC placement problem. Using ILP formulations, the authors assumed that the objective functions have a linear behavior and for optimizing these models. They used traditional ILP solvers or proposed heuristics to deal with bigger scenarios. Some recent work used machine learning strategies also, such as deep learning and reinforcement learning [57, 58, 84, 98]. In case of deep learning, the models can learn intricate structures in high-dimensional data, and can be used to make predictions based on new data [121]. Using reinforcement learning, an agent can be created to learn how to map situations to actions in order to maximize a numerical reward signal [122]. Therefore, these approaches can be used to create models for different types of SFC placement problems, in different network scenarios, for mono and multi-objective optimization.

Despite the existence of many works in the literature, there are still several open research challenges that deserve our attention. For instance, [29] highlight the need for implementing their solutions in a real testbed, and from our systematic review, one could note that there is a current gap once the most works have evaluated their proposals by using simulators. The design and implementation of a testbed is a time demanding task and requires deep technical knowledge, but it brings more realist results, highlighting the challenges and providing experience to implement the solution in a real scenario.

Several technologies have emerged to implement SFC management system prototypes, such as Open Source MANO,6 OpenStack Tacker,7 ONAP,8 Kubernetes, among others. However, there are emerging technologies that can be used in future

6 https://osm.etsi.org/.
7 https://wiki.openstack.org/wiki/Tacker,
8 https://www.onap.org/.
implementations. An example is the Programming Protocol-Independent Packet Processors (P4), which allows to describe the pipeline of packets in the network, including the parsing, match-action, and deparsing [123]. The latest version of P4 language [124] is independent of the platform where it will be executed, both of which can evolve independently, and facilitating the operability of the P4 language. In other words, the P4 allows programming the network behavior for different programmable devices. These aspects are relevant for SFC implementations and prototypes to control and adapt the traffic among the VNFs.

Taking into account the system model formulation considered in the works, it is common to consider generic aspects regarding the computing and link resources of the physical infrastructure. For instance, several works considered the processing resources just as “CPU”. However, depending on the scenario, several processor types can be available in the physical nodes. A good example is the solution proposed by Jin et al. [40] that is focused on SFC placement for mobile games. As defined by the authors, the online game VNFs can require CPU and/or GPUs to execute the tasks, and these processor types have different costs. Therefore, this differentiation has an impact on the overall SFC cost and is relevant to be considered.

Another example is about the delays considered in the SFC. The majority of work considered only delays about the links and the processing delay by VNFs. However, other delays can be present depending on the scenario. An interesting example is the solution presented in [73] for SFC placement in 5G networks. As the main goal of the paper is to minimize the overall SFC delay, several delays specific to mobile networks are considered such as user equipment processing delay, over-the-air transmission delay, the transmission about the base stations, among others. Whether only the delay about the links were considered, a relevant part of the total delay of the SFC would be disregarded in this scenario, thus, the presented solution could not be adapted so well if it were tried in a real scenario.

7 Conclusions

This work presented a systematic review on the problem of SFC placement in distributed scenarios. In this context, the various network restrictions imposed by the increasing number of devices and the emergence of applications sensitive to such restrictions (e.g., delay, bandwidth, and energy consumption) are pointed out as the main challenges faced by operators. In highly distributed scenarios, these challenges may even be greater in the face of network fluctuations.

In this review, solutions were found whose objectives focus on optimizing operational cost, delay, resource utilization, energy consumption, and revenue, considering different distributed scenarios. Although most of these works considers the optimization of only one of these metrics, other studies have formulated their optimization problems considering two or more objectives at the same time. As for the formulation of the problem, the most common optimization framework among the studies is the integer programming technique.
With regard to problem formulation, most of the algorithms were designed using some heuristic. Besides, dynamic programming and reinforcement learning were also investigated in many of the studies.

To evaluate the proposed solutions, the authors considered different scenarios, such as Fog Computing, Mobile Edge Computing, and Content Delivery Networks. However, the most popular scenario among the studies was ISP Networks. The scenario involved evaluations that took place considering different metrics. The most commonly presented metric are the network delay of the SFC allocated, the operational costs, and the use of resources for the placement of the SFCs. The acceptance rate was also one of the most popular metrics considered, allowing to measure reliability, in addition to the execution time of the proposed solutions by the studies, which allowed to measure their performance.

References

1. Asad, M., Basit, A., Qaisar, S., Ali, M.: Beyond 5g: Hybrid end-to-end quality of service provisioning in heterogeneous iot networks. IEEE Access 8, 192320–192338 (2020)
2. Sun, G., Li, Y., Liao, D., Chang, V.: Service function chain orchestration across multiple domains: a full mesh aggregation approach. IEEE Trans. Netw. Service Manag. 15(3), 1175–1191 (2018)
3. Sahaf, S., Tavernier, W., Czentye, J., Sonkoly, B., Sköldström, P., Jocha, D., Garay, J.: Scalable architecture for service function chain orchestration. In: 2015 Fourth European Workshop on Software Defined Networks, pp. 19–24. IEEE (2015)
4. Souza, R., Santos, M., Fernandes, S.: Importance measures for nfv data center: an availability evaluation. In: Anais do V Workshop Pré-IETF. SBC (2018)
5. Sun, G., Chen, Z., Yu, H., Du, X., Guizani, M.: Online parallelized service function chain orchestration in data center networks. IEEE Access 7, 100147–100161 (2019)
6. Savi, M., Tornatore, M., Verticale, G.: Impact of processing-resource sharing on the placement of chained virtual network functions. IEEE Trans. Cloud Comput. (2019). https://doi.org/10.1109/TCC.2019.2914387
7. Mijumbi, R., Serrat, J., Gorricho, J.L., Bouten, N., De Turck, F., Boutaba, R.: Network function virtualization: state-of-the-art and research challenges. IEEE Commun. Surv. Tutor. 18(1), 236–262 (2015)
8. Bonfim, M.S., Dias, K.L., Fernandes, S.F.: Integrated nfv/sdn architectures: a systematic literature review. ACM Comput. Surv. 51(6), 1–39 (2019)
9. Fan, J., Guan, C., Zhao, Y., Qiao, C.: Availability-aware mapping of service function chains. In: IEEE INFOCOM 2017-IEEE Conference on Computer Communications, pp. 1–9. IEEE (2017)
10. Zhang, C., Wang, X., Dong, A., Zhao, Y., Li, F., Huang, M.: The intelligent multi-domain service function chain deployment: architecture, challenges and solutions. Int. J. Commun. Syst. 34, 1 (2021)
11. Barroso, L.A., Höflzle, U.: The datacenter as a computer: an introduction to the design of warehouse-scale machines. Synth. Lectures Comput. Architect. 4(1), 1–108 (2009)
12. Kuo, T.W., Liou, B.H., Lin, K.C.J., Tsai, M.J.: Deploying chains of virtual network functions: on the relation between link and server usage. IEEE/ACM Trans. Netw. 26(4), 1562–1576 (2018)
13. Nan, Y., Li, W., Bao, W., Delicato, F.C., Pires, P.F., Zomaya, A.Y.: Cost-effective processing for delay-sensitive applications in cloud of things systems. In: 2016 IEEE 15th international symposium on network computing and applications (NCA), pp. 162–169. IEEE (2016)
14. Zhao, T., Zhou, S., Guo, X., Zhao, Y., Niu, Z.: A cooperative scheduling scheme of local cloud and internet cloud for delay-aware mobile cloud computing. In: 2015 IEEE Globecom Workshops (GC Wkshps), pp. 1–6. IEEE (2015)
15. Chen, H., Wang, X., Zhao, Y., Song, T., Wang, Y., Xu, S., Li, L.: Mosc: a method to assign the outsourcing of service function chain across multiple clouds. Comput. Netw. 133, 166–182 (2018)
16. Liu, Y., Zhang, H., Chang, D., Hu, H.: Gdm: a general distributed method for cross-domain service function chain embedding. IEEE Trans. Netw. Service Manag. 17(3), 1446–1459 (2020)
17. Bhamare, D., Jain, R., Samaka, M., Erbad, A.: A survey on service function chaining. J. Netw. Comput. Appl. 75, 138–155 (2016)
18. Rotos, C., King, D., Farshad, A., Bird, J., Fawcett, L., Georgalas, N., Gunkel, M., Shiomoto, K., Wang, A., Mauthe, A., et al.: Network service orchestration standardization: a technology survey. Comput. Standards Interfaces 54, 203–215 (2017)
19. Hantouti, H., Benamar, N., Taleb, T., Laghrissi, A.: Traffic steering for service function chaining. IEEE Commun. Surv. Tutor. 21(1), 487–507 (2018)
20. Souza, R., Dias, K., Fernandes, S.: Nfv data centers: a systematic review. IEEE Access 8, 51713–51735 (2020)
21. Schardong, F., Nunes, I., Schaeffer-Filho, A.: Nfv resource allocation: a systematic review and taxonomy of vnf forwarding graph embedding. Comput. Netw. 185, 107726 (2020)
22. Mirjalily, G., Zhiquan, L.: Optimal network function virtualization and service function chaining: a survey. Chin. J. Electron. 27(4), 704–717 (2018)
23. John, W., Pentikousis, K., Agapiou, G., Jacob, E., Kind, M., Manzalini, A., Risso, F., Staessens, D., Steinert, R., Meirosu, C.: Research directions in network service chaining. In: 2013 IEEE SDN for Future Networks and Services (SDN4FNS), pp. 1–7. IEEE (2013)
24. Cardoso, J., Barros, A., May, N., Kylau, U.: Towards a unified service description language for the internet of services: requirements and first developments. In: 2010 IEEE International Conference on Services Computing, pp. 602–609. IEEE (2010)
25. Sun, L., Dong, H., Ashraf, J.: Survey of service description languages and their issues in cloud computing. In: 2012 Eighth International Conference on Semantics, Knowledge and Grids, pp. 128–135. IEEE (2012)
26. Ghazouani, S., Slimani, Y.: Towards a standardized cloud service description based on usdl. J. Syst. Softw. 132, 1–20 (2017)
27. Mehraghdam, S., Karl, H.: Placement of services with flexible structures specified by a yang data model. In: 2016 IEEE NetSoft Conference and Workshops (NetSoft), pp. 184–192. IEEE (2016)
28. Katsaros, G., Menzel, M., Lenk, A., Revelant, J.R., Skipp, R., Eberhardt, J.: Cloud application portability with tosca, chef and openstack. In: 2014 IEEE International Conference on Cloud Engineering, pp. 295–302. IEEE (2014)
29. Yang, S., Li, F., Trajanovski, S., Chen, X., Wang, Y., Fu, X.: Delay-aware virtual network function placement and routing in edge clouds. IEEE Trans. Mob. Comput. 99, 1 (2019)
30. Ren, W., Sun, Y., Luo, H., Obaidat, M.S.: A new scheme for iot service function chains orchestration in sdn-iot network systems. IEEE Syst. J. 13(4), 4081–4092 (2019)
31. Kouah, R., Alleg, A., Laraba, A., Ahmed, T.: Energy-aware placement for iot-service function chain. In: 2018 IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), pp. 1–7. IEEE (2018)
32. Kiji, N., Sato, T., Shinkuma, R., Oki, E.: Virtual network function placement and routing for multi-cast service chaining using merged paths. Opt. Switch. Netw. 36, 100554 (2020)
33. Jiao, S., Zhang, X., Yu, S., Song, X., Xu, Z.: Joint virtual network function selection and traffic steering in telecom networks. In: GLOBECOM 2017-2017 IEEE Global Communications Conference, pp. 1–7. IEEE (2017)
34. Alameddine, H.A., Assi, C., Tushar, M.H.K., Yu, J.Y.: Low-latency service schedule orchestration in nfv-based networks. In: 2019 IEEE Conference on Network Softwarization (NetSoft), pp. 378–386. IEEE (2019)
35. Riera, J.F., Escalona, E., Batalle, J., Grasa, E., Garcia-Espin, J.A.: Virtual network function scheduling: concept and challenges. In: 2014 International Conference on Smart Communications in Network Technologies (SaCoNet), pp. 1–5. IEEE (2014)
36. Mijumbi, R., Serrat, J., Gorricho, J.L., Bouten, N., De Turck, F., Davy, S.: Design and evaluation of algorithms for mapping and scheduling of virtual network functions. In: Proceedings of the 2015 1st IEEE conference on network softwarization (NetSoft), pp. 1–9. IEEE (2015)
37. Qu, L., Assi, C., Shaban, K.: Delay-aware scheduling and resource optimization with network function virtualization. IEEE Trans. Commun. 64(9), 3746–3758 (2016)
38. Coutinho, E.F., de Carvalho Sousa, F.R., Rego, P.A.L., Gomes, D.G., de Souza, J.N.: Elasticity in cloud computing: a survey. Ann. Telecommun. 70(7–8), 289–309 (2015)
39. Fulber-Garcia, V., Duarte, E.P., Jr., Huff, A., dos Santos, C.R.: Network service topology: Formalization, taxonomy and the custom specification model. Comput. Netw. 178, 107337 (2020)
40. Jin, H., Zhu, X., Zhao, C.: Computation offloading optimization based on probabilistic SFC for mobile online gaming in heterogeneous network. IEEE Access 7, 52168–52180 (2019)
41. Pei, J., Hong, P., Xue, K., Li, D.: Efficiently embedding service function chains with dynamic virtual network function placement in geo-distributed cloud system. IEEE Trans. Parallel Distrib. Syst. 30(10), 2179–2192 (2018)
42. Bouet, M., Leguay, J., Combe, T., Conan, V.: Cost-based placement of V DPI functions in NFV infrastructures. Int. J. Netw. Manag. 25(6), 490–506 (2015)
43. Bari, F., Chowdhury, S.R., Ahmed, R., Boutaba, R., Duarte, O.C.M.B.: Orchestrating virtualized network functions. IEEE Trans. Netw. Service Manag. 13(4), 725–739 (2016)
44. Liu, Y., Pei, J., Hong, P., Li, D.: Cost-efficient virtual network function placement and traffic steering. In: ICC 2019-2019 IEEE International Conference on Communications (ICC), pp. 1–6. IEEE (2019)
45. Noghani, K.A., Kassler, A., Taheri, J.: On the cost-optimality trade-off for service function chain reconfiguration. In: 2019 IEEE 8th International Conference on Cloud Networking (CloudNet), pp. 1–6. IEEE (2019)
46. Feng, H., Llorca, J., Tulino, A.M., Molisch, A.F.: Optimal dynamic cloud network control. IEEE/ACM Trans. Netw. 26(5), 2118–2131 (2018)
47. He, W., Li, W., Xu, S.: A Lyapunov drift-plus-penalty-based multi-objective optimization of energy consumption, construction period and benefit. KSCE J. Civ. Eng. 146, 1–14 (2020)
48. Neely, M.J.: Queue stability and probability 1 convergence via Lyapunov optimization. arXiv preprint arXiv:1008.3519 (2010)
49. Tang, L., Yang, H., Ma, R., Hu, L., Wang, W., Chen, Q.: Queue-aware dynamic placement of virtual network functions in 5G access network. IEEE Access 6, 44291–44305 (2018)
50. Gupta, L., Samaka, M., Jain, R., Erbad, A., Bhamare, D., Metz, C.: Colap: A predictive framework for service function chain placement in a multi-cloud environment. In: 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1–9. IEEE (2017)
51. Yao, Y., Guo, S., Li, P., Liu, G., Zeng, Y.: Forecasting assisted VNFR scaling in NFV-enabled networks. Computer Networks 168, 107040 (2020)
52. Jahromi, N.T., Kianpisheh, S., Glitho, R.H.: Online VNFR placement and chaining for value-added services in content delivery networks. In: 2018 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), pp. 19–24. IEEE (2018)
53. Hejja, K., Hesselbach, X.: Offline and online power aware resource allocation algorithms with migration and delay constraints. Comput. Netw. 158, 17–34 (2019)
54. Mohamad, A., Hassanein, H.S.: On demonstrating the gain of SFC placement with VNF sharing at the edge. In: 2019 IEEE Global Communications Conference (GLOBECOM), pp. 1–6. IEEE (2019)
55. Gurobi Optimization, I.: Gurobi optimizer reference manual. http://www.gurobi.com (2018)
56. Guo, H., Wang, Y., Li, Z., Qiu, X., An, H., Yuan, N., et al.: Cost-aware placement and chaining of service function chain with VNF instance sharing. In: NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, pp. 1–8. IEEE (2020)
57. Luo, Z., Wu, C., Li, Z., Zhou, W.: Scaling geo-distributed network function chains: a prediction and learning framework. IEEE J. Select. Areas Commun. 37(8), 1838–1850 (2019)
58. Guo, S., Dai, Y., Xu, S., Qiu, X., Qi, F.: Trusted cloud-edge network resource management: DrL-driven service function chain orchestration for IoT. IEEE Internet Things J. 7, 7 (2019)
59. Nguyen, D.T., Pham, C., Nguyen, K.K., Cheriet, M.: Placement and chaining for run-time IoT service deployment in edge-cloud. IEEE Trans. Netw. Service Manag. 17(1), 459–472 (2019)
60. Fang, L., Zhang, X., Sood, K., Wang, Y., Yu, S.: Reliability-aware virtual network function placement in carrier networks. J. Netw. Comput. Appl. 154, 102536 (2020)
61. Tomassilli, A., Girouire, F., Huin, N., Pérennes, S.: Provably efficient algorithms for placement of service function chains with ordering constraints. In: IEEE INFOCOM 2018-IEEE Conference on Computer Communications, pp. 774–782. IEEE (2018)
62. Tastevin, N., Obadia, M., Bouet, M.: A graph approach to placement of service functions chains. In: 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM), pp. 134–141. IEEE (2017)
63. Chamberland, S., St-Hilaire, M., Pierre, S.: On the point-of-presence optimization problem in IP networks. Can. J. Elec. Comput. Eng. 30(3), 137–143 (2005)
64. Chamberland, S.: Point of presence design in internet protocol networks with performance guarantees. Comput. Oper. Res. 32(12), 3247–3264 (2005)
65. Tashtarian, F., Zhani, M.F., Fatemipour, B., Yazdani, D.: Codec: a cost-effective and delay-aware sfc deployment. IEEE Trans. Netw. Service Manag. 17, 793 (2019)
66. Alleg, A., Ahmed, T., Mosbah, M., Riggio, R., Boutaba, R.: Delay-aware vnf placement and chaining based on a flexible resource allocation approach. In: 2017 13th International Conference on Network and Service Management (CNSM), pp. 1–7. IEEE (2017)
67. Cai, J., Huang, Z., Luo, J., Liu, Y., Zhao, H., Liao, L.: Composing and deploying parallelized service function chains. J. Netw. Comput. Appl. 163, 102637 (2020)
68. Luizelli, M.C., Bays, L.R., Buriol, L.S., Barcellos, M.P., Gaspery, L.P.: Piecing together the nfv provisioning puzzle: Efficient placement and chaining of virtual network functions. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 98–106. IEEE (2015)
69. Sheoran, A., Sharma, P., Fahmy, S., Saxena, V.: Contain-ed: an nfv micro-service system for containing e2e latency. ACM SIGCOMM Comput. Commun. Rev. 47(5), 54–60 (2017)
70. Xu, Q., Gao, D., Li, T., Zhang, H.: Low latency security function chain embedding across multiple domains. IEEE Access 6, 14474–14484 (2018)
71. Martín-Pérez, J., Bernardos, C.J.: Multi-domain vnf mapping algorithms. In: 2018 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), pp. 1–6. IEEE (2018)
72. Zamani, A., Sharifian, S.: A novel approach for service function chain (sfc) mapping with multiple sfc instances in a fog-to-cloud computing system. In: 2018 4th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS), pp. 48–52. IEEE (2018)
73. Subramanya, T., Harutyunyan, D., Riggio, R.: Machine learning-driven service function chain placement and scaling in mec-enabled 5g networks. Comput. Netw. 166, 106980 (2020)
74. Bhame, D., Samaka, M., Erbad, A., Jain, R., Gupta, L., Chan, H.A.: Optimal virtual network function placement in multi-cloud service function chaining architecture. Comput. Commun. 102, 1–16 (2017)
75. Bhame, D., Erbad, A., Jain, R., Zolanvari, M., Samaka, M.: Efficient virtual network function placement strategies for cloud radio access networks. Comput. Commun. 127, 50–60 (2018)
76. Forney, G.D.: The viterbi algorithm. Proc. IEEE 61(3), 268–278 (1973)
77. Sun, G., Li, Y., Li, Y., Liao, D., Chang, V.: Low-latency orchestration for workflow-oriented service function chain in edge computing. Future Gen. Comput. Syst. 85, 116–128 (2018)
78. Santos, J., Wauters, T., Volckaert, B., De Turck, F.: Towards delay-aware container-based service function chaining in fog computing. In: NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, pp. 1–9. IEEE (2020)
79. Xu, Y., Kafle, V.P.: A mathematical model and dynamic programming based scheme for service function chain placement in nfv. IEICE Trans. Inform. Syst. 102(5), 942–951 (2019)
80. Li, D., Hong, P., Xue, K., Pei, J.: Virtual network function placement and resource optimization in nfv and edge computing enabled networks. Comput. Netw. 152, 12–24 (2019)
81. Shang, X., Li, Z., Yang, Y.: Placement of highly available virtual network functions through local rerouting. In: 2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), pp. 80–88. IEEE (2018)
82. Khebbache, S., Hadji, M., Zeghlache, D.: Virtualized network functions chaining and routing algorithms. Comput. Netw. 114, 95–110 (2017)
83. Carpio, F., Bziuk, W., Jukan, A.: Replication of virtual network functions: optimizing link utilization and resource costs. In: 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 521–526. IEEE (2017)
84. Li, G., Feng, B., Zhou, H., Zhang, Y., Sood, K., Yu, S.: Adaptive service function chaining mappings in 5g using deep q-learning. Comput. Commun. 152, 305–315 (2020)
85. Song, X., Zhang, X., Yu, S., Jiao, S., Xu, Z.: Resource-efficient virtual network function placement in operator networks. In: GLOBECOM 2017-2017 IEEE Global Communications Conference, pp. 1–7. IEEE (2017)
86. Soualah, O., Mechtri, M., Ghribi, C., Zeghlache, D.: Energy efficient algorithm for vnf placement and chaining. In: 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), pp. 579–588. IEEE (2017)
87. ISG, N.: Network functions virtualisation (nfv)-virtualisation requirements. ETSI Technical Report (2013)
88. Sun, G., Li, Y., Yu, H., Vasilakos, A.V., Du, X., Guizani, M.: Energy-efficient and traffic-aware service function chaining orchestration in multi-domain networks. Future Gen. Comput. Syst. 91, 347–360 (2019)

89. Xu, Z., Zhang, X., Yu, S., Zhang, J.: Energy-efficient virtual network function placement in telecom networks. In: 2018 IEEE International Conference on Communications (ICC), pp. 1–7. IEEE (2018)

90. Zhang, X., Xu, Z., Fan, L., Yu, S., Qu, Y.: Near-optimal energy-efficient algorithm for virtual network function placement. IEEE Trans. Cloud Comput. 99, 1–1 (2019)

91. Saha, S., Sarkar, J., Dwivedi, A., Dwivedi, N., Narasimhamurthy, A.M., Roy, R.: A novel revenue optimization model to address the operation and maintenance cost of a data center. J. Cloud Comput. 5(1), 1–23 (2016)

92. Sheikholeslami, F., Navimipour, N.J.: Service allocation in the cloud environments using multi-objective particle swarm optimization algorithm based on crowding distance. Swarm Evol. Comput. 35, 53–64 (2017)

93. Badshah, A., Ghani, A., Shamshirband, S., Chronopoulos, A.T.: Optimising infrastructure as a service provider revenue through customer satisfaction and efficient resource provisioning in cloud computing. IET Commun. 13(18), 2913–2922 (2019)

94. Badshah, A., Ghani, A., Shamshirband, S., Aceto, G., Pescaïre, A.: Performance-based service-level agreement in cloud computing to optimise penalties and revenue. IET Commun. 14(7), 1102–1112 (2020)

95. Mechtri, M., Ghribi, C., Soualah, O., Zeghlache, D.: Nfv orchestration framework addressing sfc challenges. IEEE Commun. Mag. 55(6), 16–23 (2017)

96. Ma, Y., Liang, W., Xu, Z., Guo, S.: Profit maximization for admitting requests with network function services in distributed clouds. IEEE Trans. Parallel Distrib. Syst. 30(5), 1143–1157 (2018)

97. Xie, Y., Wang, S., Dai, Y.: Revenue-maximizing virtualized network function chain placement in dynamic environment. Future Gen. Comput. Syst. (2020). https://doi.org/10.1016/j.future.2020.03.011

98. Li, G., Zhou, H., Feng, B., Zhang, Y., Yu, S.: Efficient provision of service function chains in overlay networks using reinforcement learning. IEEE Trans. Cloud Comput. (2019). https://doi.org/10.1109/TCC.2019.2961093

99. Li, G., Zhou, H., Feng, B., Li, G.: Context-aware service function chaining and its cost-effective orchestration in multi-domain networks. IEEE Access 6, 34976–34991 (2018)

100. Cappanera, P., Paganelli, F., Paradiso, F.: Vnf placement for service chaining in a distributed cloud environment with multiple stakeholders. Comput. Commun. 133, 24–40 (2019)

101. Chen, Y.T., Liao, W.: Mobility-aware service function chaining in 5g wireless networks with mobile edge computing. In: ICC 2019-2019 IEEE International Conference on Communications (ICC), pp. 1–6. IEEE (2019)

102. Cui, Y., Geng, Z., Zhu, Q., Han, Y.: Multi-objective optimization methods and application in energy saving. Energy 125, 681–704 (2017)

103. Bouten, N., Mijumbi, R., Serrat, J., Famaey, J., Latrè, S., De Turck, F.: Semantically enhanced mapping algorithm for affinity-constrained service function chain requests. IEEE Trans. Netw. Service Manag. 14(2), 317–331 (2017)

104. Gupta, L., Jain, R., Erbad, A., Bhamare, D.: The p-art framework for placement of virtual network services in a multi-cloud environment. Comput. Commun. 139, 103–122 (2019)

105. Zhang, C., Wang, X., Dong, A., Zhao, Y., He, Q., Huang, M.: Energy efficient network service deployment across multiple sdn domains. Comput. Commun. 151, 449–462 (2020)

106. Troia, S., Cibari, A., Alvizù, R., Maier, G.: Dynamic programming of network slices in software-defined metro-core optical networks. Opt. Switch. Netw. 36, 100551 (2020)

107. Harutyunyan, D., Fedrizzi, R., Shahriar, N., Boutaba, R., Riggio, R.: Orchestration end-to-end slices in 5g networks. In: 2019 15th International Conference on Network and Service Management (CNSM), pp. 1–9. IEEE (2019)

108. Harutyunyan, D., Shahriar, N., Boutaba, R., Riggio, R.: Latency-aware service function chain placement in 5g mobile networks. In: 2019 IEEE Conference on Network Softwarization (NetSoft), pp. 133–141. IEEE (2019)

109. Jin, H., Jin, Y., Lu, H., Zhao, C., Peng, M.: Nfv and sfc: a case study of optimization for virtual mobility management. IEEE J. Select. Areas Commun. 36(10), 2318–2332 (2018)
110. Shang, X., Liu, Z., Yang, Y.: Network congestion-aware online service function chain placement and load balancing. In: Proceedings of the 48th International Conference on Parallel Processing, pp. 1–10 (2019)

111. Eramo, V., Miucci, E., Ammar, M., Lavacca, F.G.: An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures. IEEE/ACM Trans. Network. 25(4), 2008–2025 (2017)

112. Siasi, N., Jaesim, A., Ghani, N.: Tabu search for efficient service function chain provisioning in fog networks. In: 2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC), pp. 145–150. IEEE (2019)

113. Tajiki, M.M., Shojafar, M., Akbari, B., Salsano, S., Conti, M., Singhal, M.: Joint failure recovery, fault prevention, and energy-efficient resource management for real-time sfc in fog-supported sdn. Comput. Netw. 162, 106850 (2019)

114. Son, J., Buyya, R.: Latency-aware virtualized network function provisioning for distributed edge clouds. J. Syst. Softw. 152, 24–31 (2019)

115. Zhao, D., Sun, G., Liao, D., Xu, S., Chang, V.: Mobile-aware service function chain migration in cloud-fog computing. Future Gen. Comput. Syst. 96, 591–604 (2019)

116. Chiosi, M., BT, C.D., Peter, W., Centurylink, A.R., James, F., Michael, B., Waqar, K., Michael, F.: Network functions virtualization—introductory white paper issue I network functions virtualization an introduction, benefits, enablers, challenges & call for action contributing organisations & authors (2012)

117. Condoluci, M., Mahmoodi, T.: Softwarization and virtualization in 5G mobile networks: benefits, trends and challenges. Comput. Netw. 146, 65–84 (2018)

118. Stiawan, D.: Personal data protection and liability of internet service provider a comparative. Int. J. Electr. Comput. Eng. 9(4), 3175–3184 (2019)

119. Hong, D.K., Ma, Y., Banerjee, S., Mao, Z.M.: Incremental deployment of sdn in hybrid enterprise and isp networks. In: Proceedings of the Symposium on SDN Research, pp. 1–7 (2016)

120. Yu, H., Lee, H., Jeon, H.: What is 5G? Emerging 5G mobile services and network requirements. Sustainability 9(10), 1848 (2017)

121. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature 521(7553), 436–444 (2015)

122. Sutton, R.S., Barto, A.G.: Reinforcement learning: an introduction. MIT, New York (2018)

123. Harkous, H., Jarschel, M., He, M., Priest, R., Kellerer, W.: Towards understanding the performance of p4 programmable hardware. In: 2019 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), pp. 1–6. IEEE (2019)

124. Consortium, P.L.: P4 language and related specifications. https://p4.org/specs/ (2021)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Guto Leoni Santos is B.Sc. in Information System from Universidade de Pernambuco and obtained his M.Sc. degree in Computer Science from Universidade Federal de Pernambuco (UFPE). Currently, is a Ph.D. student of Computer Science at UFPE and has taken part in the GPRT since 2016. He is taking part in GPRT and his research interests include: cloud Computing, Performance Evaluation, Internet of Things, Fog computing, and Artificial Intelligence, Machine Learning, and Deep Learning.

Diego de Freitas Bezerra is B.Sc. in Computer Science and obtained his M.Sc. degree in Applied Informatics from Universidade Federal Rural de Pernambuco (UFRPE). Currently, is a Ph.D. student in Computer Science at Universidade Federal de Pernambuco (UFPE). He has experience in Computer Science, with emphasis on Methodology and Computer Techniques, working mainly in the following subjects: Discrete Event Systems, Machine Learning, and Internet of Things.

Élisson da Silva Rocha has a master degree in Computer Engineering (PPGEC) from the Universidade de Pernambuco (UPE). Graduated in Information System from Universidade de Pernambuco – campus Caruaru (2017). He conducts research at GPRT of Universidade Federal de Pernambuco (UFPE) and has experience in Computer Science, focusing on Computer Systems, acting on the following subjects: Computational intelligence, neural networks, cloud computing, distributed systems.
Leylane Ferreira is a Ph.D. student of Computer Science at Universidade Federal de Pernambuco (UFPE) and member of GPRT since 2018. Her current research interests include Cloud and Fog Computing, Smart Cities and Optimization Algorithms.

André Luís Cavalcanti Moreira received his Ph.D. of Computer Science from the Universidade Federal de Pernambuco (UFPE). His research topic is in self organization of cloud networks and adaptation of CDN provisioning algorithms. Currently, he is involved in a research project in a platform for clouds at GPRT.

Glauco Estácio Gonçalves is a professor at Federal University of Pará (UFPA) since 2013. He received his Ph.D. degree in Computer Science from the Universidade Federal de Pernambuco (UFPE) in 2012. His research interests include: Performance Evaluation of Networked Systems; cloud Computing; and Optimization Algorithms for Resource Allocation.

Maria Valéria Marquezini received her Doctor of Science (Physics) degree in 1995 from the State University of Campinas- UNICAMP, Brazil. Between 1995-97 she worked as a post-doctoral researcher in the Materials Science division of the Lawrence Berkeley National Laboratory and from 1997-99 she worked as an associate researcher at the Physics Institute, UNICAMP, conducting research activities in the field of electronic processes in semiconductors. Since 2000 she has been working in the research department of Ericsson in Brazil and has been engaged in many research projects in the telecommunications domain.

Ákos Recse holds a BSc in Computer Science Engineering and Masters in Business Information Systems from Budapest University of Technology and Economics. Since then he is working on his Ph.D. at ELTE related to IoT and distributed cloud. His main focus is on VNF orchestration in distributed cloud scenarios. He has six years of working experience in the cloud research area of Ericsson as a researcher intern. Through these years he acquired experience working both in small agile design teams and in pan-European projects.

Amardeep Mehta works as a researcher in Cloud Systems and Platforms at Ericsson Research, Stockholm. He received his M.Sc. in Computer Science from Uppsala University and Ph.D. in Computing Science from Umeå University, Sweden. His research interests include workload analysis and resource management problems for distributed systems.

Judith Kelner received her Ph.D. from the Computing Laboratory at the University of Kent at Canterbury, UK in 1993. She has been a Full Professor at Universidade Federal de Pernambuco (UFPE), since 1979. Currently she leads the GRVM team as well as coordinates a number of research projects in the areas of multimedia systems, design of virtual and augmented reality applications, and smart communication devices.

Djamel Sadok received his Ph.D. of Computer Science at the University of Kent at Canterbury, UK in 1990. He has been a member of staff at Universidade Federal de Pernambuco (UFPE) since 1993. His research interests include communication systems, access networks, security, cloud computing and traffic classification. Currently he leads the GPRT team as well coordinates a number of research projects.

Patricia Takako Endo received her Ph.D. of Computer Science from the Universidade Federal de Pernambuco (UFPE), Brazil in 2014. She has been a professor at Universidade Federal de Pernambuco (UPE), Brazil since 2009; and a researcher at Research in Networks and Telecommunication Group (GPRT) since 2009. Her current research interests are: cloud and fog computing, and resource management.
Authors and Affiliations

Guto Leoni Santos1 · Diego de Freitas Bezerra1 · Élisson da Silva Rocha1 · Leylane Ferreira1 · André Luis Cavalcanti Moreira1 · Glauco Estácio Gonçalves2 · Maria Valéria Marquezini3 · Ákos Recse4 · Amardeep Mehta5 · Judith Kelner1 · Djamel Sadok1 · Patricia Takako Endo6

Guto Leoni Santos
gls4@cin.ufpe.br

Diego de Freitas Bezerra
diego.bezerra@gprt.ufpe.br

Élisson da Silva Rocha
elisson.rocha@gprt.ufpe.br

Leylane Ferreira
leylane.silva@gprt.ufpe.br

André Luis Cavalcanti Moreira
andre@gprt.ufpe.br

Glauco Estácio Gonçalves
glaucogoncalves@ufpa.br

Maria Valéria Marquezini
maria.marquezini@ericsson.com

Ákos Recse
akosrecse@inf.elte.hu

Amardeep Mehta
amardeep.mehta@ericsson.com

Judith Kelner
jk@gprt.ufpe.br

Djamel Sadok
jamel@gprt.ufpe.br

1 Universidade Federal de Pernambuco, Recife, Brazil
2 Universidade Federal do Pará, Belém, Brazil
3 Ericsson Research, Indaiatuba, Brazil
4 Eötvös Loránd University, Budapest, Hungary
5 Ericsson Research, Kista, Sweden
6 Universidade de Pernambuco, Recife, Brazil