applied sciences

MDPI

Article

A Genetic Algorithm for VNF Provisioning in NFV-Enabled Cloud/MEC RAN Architectures

Lidia Ruiz [1,*], Ramón J. Durán [1,*], Ignacio de Miguel [1], Pouria S. Khodashenas [2], Jose-Juan Pedreno-Manresa [3], Noemí Merayo [1], Juan C. Aguado [1], Pablo Pavon-Marino [3], Shuaib Siddiqui [2], Javier Mata [1], Patricia Fernández [1], Rubén M. Lorenzo [1] and Evaristo J. Abril [1]

[1] Optical Communications Group. Universidad de Valladolid. Paseo de Belén, 15, 47011 Valladolid, Spain; ignacio.miguel@tel.uva.es (I.d.M.); noemer@tel.uva.es (N.M.); jaguado@tel.uva.es (J.C.A.); javiermatag@gmail.com (J.M.); patfer@tel.uva.es (P.F.); rublor@tel.uva.es (R.M.L.); ejad@tel.uva.es (E.J.A.)

[2] i2CAT Foundation, C/Gran Capità, 2, 08034 Barcelona, Spain; pouria.khodashenas@i2cat.net (P.S.K.); shuaib.siddiqui@i2cat.net (S.S.)

[3] Telecommunication Networks Engineering Group. Universidad Politécnica de Cartagena, Cuartel de Antiguones, Plaza del Hospital 1, 30202 Cartagena, Spain; josej.pedreno@upct.es (J.-J.P.-M.); pablo.pavon@upct.es (P.P.-M.)

* Correspondence: lruiper@ribera.tel.uva.es (L.R.); rduran@tel.uva.es (R.J.D.); Tel.: +34-983-18-5557 (R.J.D.)

Received: 19 October 2018; Accepted: 10 December 2018; Published: 13 December 2018

Abstract: 5G technologies promise to bring new network and service capacities and are expected to introduce significant architectural and service deployment transformations. The Cloud-Radio Access Networks (C-RAN) architecture, enabled by the combination of Software Defined Networking (SDN), Network Function Virtualization (NFV) and Mobile Edge Computing (MEC) technologies, play a key role in the development of 5G. In this context, this paper addresses the problems of Virtual Network Functions (VNF) provisioning (VNF-placement and service chain allocation) in a 5G network. In order to solve that problem, we propose a genetic algorithm that, considering both computing resources and optical network capacity, minimizes both the service blocking rate and CPU usage. In addition, we present an algorithm extension that adds a learning stage and evaluate the algorithm performance benefits in those scenarios where VNF allocations can be reconfigured. Results reveal and quantify the advantages of reconfiguring the VNF mapping depending on the current demands. Our methods outperform previous proposals in the literature, reducing the service blocking ratio while saving energy by reducing the number of active core CPUs.

Keywords: 5G; optical network; C-RAN; NFV; SDN; VNF; VNF-Provisioning; genetic algorithm; network planning; VNF Scaling

1. Introduction

The next generation of mobile broadband communications, 5G, is expected to bring unique network and service capabilities. It will offer low latency (<1 ms end-to end service), multi-tenancy, high capacity, high-speed communications (≈1 Gbps data rate), resource virtualization, coordinated automation or heterogeneous technology convergence among other functionalities. Moreover, supporting the ever-increasing number of connected devices (up to 100-fold), offering a variety of services as online gaming, video streaming or Voice over IP (VoIP) while reducing capital (CAPEX) and operational costs (OPEX) and energy consumption are some of the challenges that emerging 5G technologies will have to overcome [1].

In order to support the increasing number of connected devices, 5G will require transport networks to cope with enormous amounts of traffic with minimum latency. Therefore, not only

the radio access network, but the metro and transport networks must be redefined from an architectural point of view. One of the most important enabling technologies for that redesign is the Centralized/Cloud Radio Access Networks (C-RAN) architecture.

In C-RAN, the base station (BS) functionalities are divided into two separate entities: Remote Radio Heads (RRH) and Based Band Units (BBU). BBUs can be placed in the same location as RRH, but economical efficiencies are achieved when BBUs are placed at higher hierarchy premises (together with other BBUs, increasing the flexibility and reducing costs).

In this architecture, the BBUs connect with the RRH through the so-called Mobile Fronthaul (MFH) segment and commonly implement the CPRI protocol to transport the traffic between RRHs and BBUs [2,3]. Since the fronthaul traffic between RRHs and BBUs is usually the sampled, quantized and encoded radio signal [4,5], it requires C-RAN to rely on networks with high capacity and guaranteed latencies. These restrictions constraint the location of the BBUs, e.g., to be hosted at the first aggregation site, although location at higher aggregation sites has been studied in Reference [6]. Moreover, 5G is expected to require CPRI rates significantly larger than the available bandwidth in current networks. For this reason, other functional splits, i.e., other divisions between the functionality that remains at the antenna site and the functionality deployed in the operator premises are considered to be deployed in 5G access networks [7]. In References [8,9] the BBUs are divided into two entities: Distributed Unit (DU), which provides the physical layer processes and the real-time functionalities of the network, and the Central Unit (CU), which deploys non-real-time wireless high-level protocol function. Nonetheless, C-RAN latency and capacity requirements render optical technologies as the perfect candidate for the 5G fronthaul segments [2].

On the other hand, Mobile Edge Computing (MEC) technology will also contribute to the realization of 5G. MEC is a technology that provides cloud computing services at the edge of the network, i.e., closer to the end-user. Therefore, bringing the data processing to the edge node avoids the traffic to travel long distances incurring in propagation delays and, consequently, it can help to significantly reduce latency [10]. MEC and C-RAN seem to be colliding concepts, since MEC aims to distribute the data processing among the edge nodes of the network, whereas C-RAN aims to centralize the data processing in the BBU pools in the operator premises. However, its use in conjunction in the network can enable the reduction of latency, cost and energy consumption. This is done by using virtualized BBUs, i.e., virtual functions running on multicommodity servers (i.e., those from MEC) implementing the functionality of a BBU [3]. Examples of architectures addressing that idea are the Heterogeneous Cloud Radio Access Network (H-CRAN) [11], which combines C-RAN with cloud computation techniques deployed at some nodes of the network, or the Cloud/Fog RAN architecture [12], which places Local Processing Units at the network cells to deploy cloud services and, therefore, virtual BBU (vBBU) functions, closer to the end user.

Network Function Virtualization (NFV) and Software Defined Networking (SDN) are also facilitating the realization of 5G by changing the manner in which operators deploy their end-user services. Nowadays, operators offer services like video streaming, web browsing, data transmission or voice services by executing specific tasks, known as network functions. Modern networks deploy network functions on dedicated hardware or middleboxes. Since middleboxes are costly and inflexible, operators incur an increase in capital and operational expenditures any time they deploy new and complex services. NFV is a technology that deploys network functions as software appliances called Virtual Network Functions (VNFs). VNFs are instantiated in Commercial-Off-The-Shelf (COTS) servers, which can host multiple instances of different VNFs, thus facilitating the deployment of complex network services and the efficient dimensioning of the network. Moreover, COTS hardware can be placed at data centers, Central Offices (CO) [13], but also optical network edge nodes can be NFV-enabled nodes thanks to the MEC approach. Consequently, NFV facilitates the deployment of new network services avoiding a rise in the network cost.

The traffic associated to each service must traverse a Service Chain (SC), i.e., a set of VNFs concatenated in a specific order while satisfying latency, and bandwidth requirements, as well as

computing resources and the maximum number of concurrent users per VNF requirements. Therefore, deciding where to host VNFs and which particular instances should be concatenated to set up SCs while satisfying the aforementioned requirements are some of the problems that operators face when implementing NFV. We refer to this problem as the VNF-Provisioning problem. Lastly, operators must decide whether to perform the VNF-Provisioning at peak loads, which may cause an inefficient use of resources, or take advantage of the benefits of NFV and SDN and perform periodic network re-plannings to adapt the resources to the actual traffic.

In this paper, we address the NFV-based network dimensioning problem in a 5G network to deploy services like Video Streaming, VoIP or Web browsing. To achieve this objective, we consider the implementation of Heterogeneous Cloud Radio Access Network (H-CRAN) in the network using traditional C-RAN technology (i.e., without splitting BBUs into DUs and CUs) and including MEC servers at the first aggregation site, i.e., at the Access Office (AO) serving each cell of the network, in order to deploy both VNF and vBBUs functions. We assume that RRHs connect to the virtualized BBUs deployed at the AO using optical communications, in a similar manner as [6,12]. We extend the Fog/Cloud RAN architecture in Reference [12] to benefit from the advantages of the MEC paradigm. Therefore, the AO also hosts a MEC server to deploy the virtual BBUs and other possible VNFs. We use optical technologies to connect the AO to the CO, equipped with IT resources and assume that the VNFs required to deploy the selected services can be deployed either on the MEC server at the AO or on the CO.

Firstly, we propose a genetic algorithm to solve the VNF-Provisioning problem. The algorithm (inspired on a previous work presented in Reference [14]) considers (i) the IT resources in the AO and the CO and (ii) the optical network resources to minimize the service blocking ratio and to reduce the CPU usage. Then, we extend the work by evaluating its performance in a more realistic scenario, with dynamic traffic, and by incorporating a learning technique to improve its performance. Moreover, we also demonstrate the advantages of performing a dynamic reconfiguration of the VNF provisioning in order to reduce service blocking ratio and CPU usage (and, therefore, the energy consumption).

The rest of the paper is organized as follows. Section 2 discusses some related work. Then, Section 3 details the proposed 5G architecture and the VNF-Provisioning solving algorithms. Section 4 describes the simulation scenario and analyses the performance of the proposed methods. Lastly, Section 5 presents the conclusions of this study.

2. Related Work

C-RAN technology is a promising approach to reduce latency in 5G networks and the literature has largely covered this topic since authors in Reference [2] made one of the first C-RAN system architecture proposals, suggesting two kinds of solutions: The "full centralization", in which all Layer 1, 2, and 3 Base Station functions are located in the BBU and the "partial centralization" where baseband functions are located in the RRH while the higher layer functions are located in BBU. But the development of 5G over C-RAN poses different challenges regarding the transport network and the achievement of 5G benchmarks, in particular with latency and bandwidth. In 2015, Pfeiffer [3] stated the necessity of revisiting the Common Public Radio Interface (CPRI) approach to building the MFH to support the increasing traffic while satisfying the challenges in the time and frequency domain. The author proposes moving from distributed to centralized RAN architectures to reduce CAPEX and OPEX and facilitate the implementations of radio techniques as Coordinated Multi Point (CoMP), using Passive Optical Network (PON) technologies like WDM-PON or TDM-PON to support the bandwidth requirements, in combination with split processing to relax the bandwidth and latency requirements in the fronthaul. Kani et al. [15] explored different PON technologies for the construction of the MFH to satisfy its bandwidth and latency requirements, as TDM-PON for its cost-effectiveness, TWDM-PON for its suitability in scenarios where the high traffic points dynamically move, and WDM-PON or Point-to-Point architectures for their high bandwidth. Furthermore, the scientific community is considering different functional splits to ease the bandwidth and capacity requirements of 5G over

C-RAN [7]. In References [8,9], C-RAN is divided into three entities: The RRUs, the DU and the CU. The RRUs may keep radio and some layer-1 functions, whereas the DU provides the rest of layer-1 processes and real-time functionality, as Hard Automatic Repeat request (HARQ) and Automatic Repeat request (ARQ), and the CU the non-real-time, high-level wireless protocol functionality. In this architecture, the fronthaul remains the segment connecting RRHs and DUs, while the segment between the DU and the CU is called Midhaul (MH). With this split, the shorter fronthaul, which remains the segment between RRH and DU, and the relaxed bandwidth and latency requirements favor the support of the real-time processes of the e-node.

In Reference [6], Musumeci et al. studied the impact of the latency requirements in the BBU consolidation in a C-RAN architecture. The authors considered WDM Aggregation Networks and three kinds of BBU split architectures in which the BBU were stored at the cabinet of the base station, at the CO at the first level of aggregation or at CO in further levels of aggregation, depending on the latency restrictions. Authors in Reference [11] introduced the Heterogeneous Cloud Radio Access Networks (H-CRAN), which combines cloud computing technologies and C-RAN to improve the performance of HetNets in terms of energy consumption and spectral efficiency.

Tinini et al. [12] extended the Hybrid C-RAN in Reference [11] to integrate the Fog Computing and NFV paradigms, by adding Local Processing Units to each Cell Site (CS) in which implement vBBUs. The authors also proposed an Integer Linear Programming (ILP) formulation to solve the BBU placement problem in the TWDM-PON based Fog/Cloud RAN architecture, with the objective of supporting high-traffic demands minimizing the energy consumption. In our work, we extend these architectures to create virtual BBU functions in a MEC server located at the first level aggregation site and, in this manner, combine MEC technologies and C-RAN and enable the possibility of virtualizing network functions.

The coexistence of MEC and C-RAN has also been covered in the literature. Rimal et al. [16] discussed MEC as a pathway for 5G realization, introduced different 5G-service scenarios and design challenges and proposed and explored the feasibility of MEC over Fiber Wireless Networks for different RAN scenarios, including the coexistence of MEC and C-RAN. Authors in References [17,18] studied the integration of MEC and C-RAN from a resource allocation viewpoint, proposing techniques for resource allocation to reduce the energy consumption, while in Reference [19], authors proposed different coordination techniques to reduce the end-to-end latency in C-RAN with MEC. Lastly, Blanco et al. [20] explored the role that MEC, together with SDN and NFV will play to address the challenges 5G aims to undertake. None of the studies on the integration of MEC and C-RAN explores the use of NFV to deploy specific VNFs to offer final user services, such as video streaming, VoIP, or web browsing.

The VNF placement and service chaining problem has also attracted the interest of the research community. In static scenarios, Lin et al. [21] solved the resource allocation problem with a mixed-integer program to optimally serve end-to-end requests, minimizing CAPEX and OPEX and added a method to consider limited physical resources. In Reference [13], Savi et al. studied the resource allocation and chaining problem considering the impact of IT resource-sharing among VNFs and the scalability issues that may appear when multiple SCs are deployed in a network through an ILP model. But the VNF placement problem is known to be NP-Hard and ILP models require large computation time to solve the problem for large networks [22]. Thus, heuristics and metaheuristics are employed to find sub-optimal solutions in polynomial time. For instance, authors in Reference [23] proposed a heuristic algorithm to study the impact of latency on VNF placement, showing that when the latency requirement of a service is strict, it can only be satisfied by deploying VNFs closer to the end user, in the metro/access network. Besides Carpio et al. [24] proposed a genetic algorithm for solving the VNF placement problem in Data Centers with the purpose of improving the load balancing. However, these efforts explore the VNF provisioning problem only on static scenarios.

Several studies explore the VNF resource allocation problem in dynamic scenarios. For example, Zeng et al. [25] addressed the VNF-Provisioning problem in an Inter-Data Center Elastic Optical

Network for both static network planning and online provisioning through a Mixed-Integer Linear Programming (MILP) model, with the aim of reducing costs. More recently, authors in Reference [26] proposed an online algorithm to solve the VNF placement and chaining problem across geo-distributed datacenters, minimizing the operational costs. Authors in References [27,28] explore the online VNF-Provisioning problem within datacenters. The first work proposes an algorithm to solve the VNF-Placement problem reducing provisioning costs and considering traffic rates between adjacent VNFs and server resource capacities, whereas the second study proposes an algorithm for service chain deployment and scaling that takes into consideration traffic fluctuations, to reduce deployment costs and link utilization. Rankothge et al. [29] presented a resource allocation algorithm based on Genetic Algorithms to solve the VNF placement in a Data Center minimizing the usage of IT resources. Otokura et al. [30] presented an online VNF placement problem solving method based on genetic algorithms which aims to minimize the required time to obtain feasible solutions. Moreover, authors in Reference [31] addressed the resource allocation problem in an NFV system, which aims to minimize the system cost and maximize the number of served requests. Lastly, authors in References [32,33] proposed a heuristic for online VNF placement and chaining in a realistic 5G scenario, aiming to reduce the service blocking rate, and taking into consideration computational resources and optical network capacity.

In contrast with those previous works, in this paper we propose a genetic algorithm for VNF-Provisioning in order to minimize both the service blocking ratio and the CPU usage in a 5G access network. We consider that RRHs are connected to an AO following a hybrid Cloud/MEC RAN architecture, based on the one proposed in Reference [12]. However, we extend said architecture by equipping MEC servers with different capabilities at each AO in order to deploy both virtualized BBUs and other VNFs. AOs connect to the Central Office, which hosts IT resources, via dedicated optical links. We test our proposal in both static and dynamic scenarios and propose a new version of the method by adding a simple learning capability with the purpose of improving the results in terms of service blocking ratio and CPU usage. Finally, we also demonstrate the advantages of reconfiguring the VNF provisioning in contrast with making a static planning and show the performance of our proposed method in that scenario.

3. Genetic Algorithm for VNF-Mapping in Cloud/MEC RAN-Based 5G Architectures

3.1. Network Architecture

We consider a Cloud/MEC RAN architecture based on the one proposed in Reference [12] which is shown in Figure 1. In this architecture, RRHs connect via optical networks (using dedicated links or PON) to an AO. The AO hosts a MEC server to bring cloud services and function virtualization closer to the end user. This server will host instances of VNFs, including vBBUs. Since the closest AO to the RRH hosts the vBBU attending the RRH, we reduce the physical distance between both elements, ensuring that the stringent 5G latency requirement is fulfilled. We distinguish between two kinds of AOs: HD-AO (High Demand AO) and LD-AO (Low Demand AO). HD-AO serves a higher number of average end users than LD-AO and, consequently, the MEC servers at the HD-AO will be equipped with higher computing resources than the servers at the LD-AO. That definition is also compatible with the division in macrocells and microcells in current mobile networks. Lastly, each AO can connect to the CO via dedicated optical links, optical networks featuring Wavelength Division Multiplexing (WDM) like in References [6,8,9] or even PON technology. We assume that the CO hosts IT resources to deploy VNF instances, but it will not host virtual BBUs. This architecture does not consider splitting the BBUs into DUs and CUs [8,9]. However, it is completely compatible with that architecture by simply hosting the DUs in the AOs while the non-time sensitive CU can be hosted in the CO.

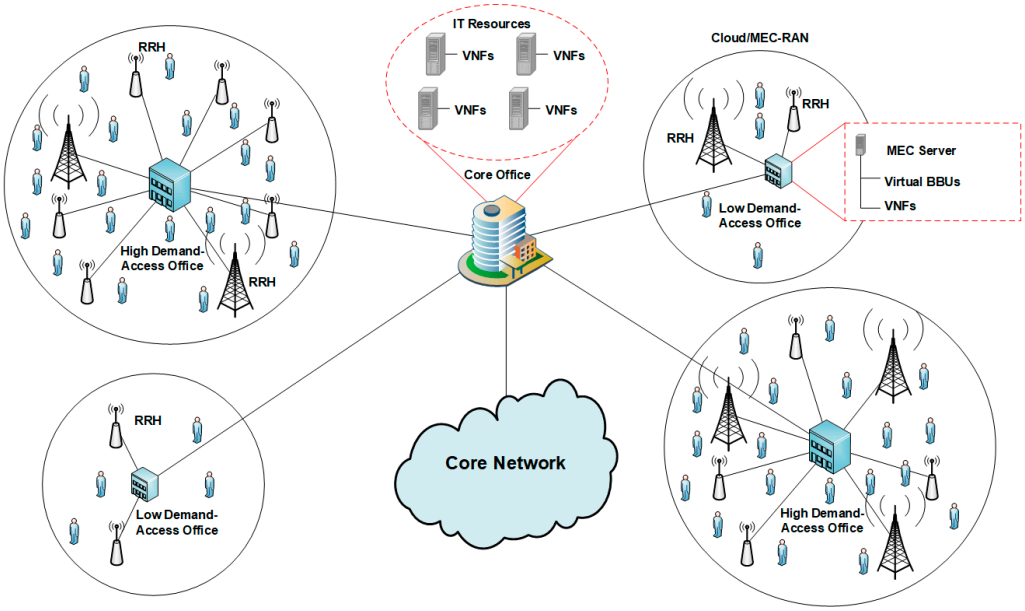


Figure 1. Proposed 5G architecture implementing Cloud/ Mobile Edge Computing (MEC) Radio Access Networks (RAN).

3.2. Problem Description

The VNF-Provisioning problem consists of two parts: The VNF-Placement (i.e., planning) and the VNF-Chaining (i.e., operation) [27,28]. The planning methods must decide how many instances of the VNFs must be allocated to each VNF-enabled host of the network [30]. The chaining methods, on the other hand, must dynamically create the necessary SCs to serve each service request from the user [31–33]. The chaining stage is closely related to the planning one, since the establishment of the SC must use the VNF currently allocated in the network, and if an SC requires not-available VNFs instances or if the network does not have enough capacity to support the establishment of the SC, the user request will fail, and the service blocking rate will increase.

The operation problem must be solved online each time that the network receives a user service request, since only at the request arrival time does the network know which type of traffic request must process and, consequently, the type of SC that must establish, and thus takes into account real service requests. However, the VNF planning problem is solved offline, based on traffic estimations, and leads to a static or a semi-static solution. In the latter case, the VNF planning problem is periodically solved, and involves the reconfiguration of the VNF mapping in order to better adapt to the current or near-term predicted traffic demand. The use of a completely dynamic method for planning is not a pragmatic approach, since setting up the VNF may take tens of seconds [34], and that increases the delay to initialize the service.

The planning algorithms receive as input the average service requests from traffic monitors, including the number and type of the requests. Moreover, they consider the MEC/Cloud RAN architecture presented in the previous section and take into account the availability of IT resources (i.e., CPU, RAM and disk) and the network bandwidth. With those inputs and knowing the requirements in terms of IT and network resources to setup a VNF in a virtual machine, the planning methods should provide the optimal (or a close-to-optimal) distribution of the VNFs at the CO and at each AO of the network for the future requests. This stage is solved offline considering two possible scenarios: A static scenario in which the placement does not change with time, and a reconfigurable scenario in which the VNF-Placement can be modified to be adapted to the current requests.

The algorithms to solve the VNF-Chaining problem receive service requests in real time. Considering the established VNFs and both the VNF and the network availability, they must provide

the SC to serve the user service request. When there are not enough resources to establish that SC, the service request is blocked.

In this paper, we propose planning methods, based on genetic algorithms, with the objective of minimizing both the service blocking rate and CPU usage (and, therefore, the energy consumption). Moreover, we also show that the performance of the networks can improve by reconfiguring the VNF mapping.

3.3. VNF-Chaining Algorithms

As we have previously mentioned, the VNF-Provisioning problem consists of the VNF-Placement and the VNF-Chaining subproblems. This paper mainly proposes VNF-Placement algorithms. However, rather than operating in a completely agnostic way, those algorithms take into count how the VNF-Chaining subproblem is solved, in order to provide better solutions. Therefore, in this section we explain several approaches for solving the VNF-Chaining subproblem.

Authors in References [32,33] proposed a method to solve the complete VNF-Provisioning problem dynamically. That method, henceforth MEC-First, operates in the following manner: When a user service request arrives, the algorithm starts the chaining process by searching available instances of the VNF at the MEC server located at the AO serving the end user. If there are not available instances of the VNF, but the MEC server has enough idle IT resources to set up the required instances, the algorithm creates them. If the algorithm cannot concatenate more VNFs hosted in the MEC server at the local AO, due to either lack of instances or to lack of IT resources, and there is enough available network bandwidth between the local AO and the CO, the chaining process continues at the CO. In the CO the algorithm repeats the same procedure, that is, it tries to utilize existing VNF instances and, if unable, tries to create new ones. If the algorithm cannot utilize either the existing VNF instances or the IT resources of the CO, it will not look for existing resources back at any local MEC server (i.e., there are no loops or backtracking). Finally, the request is served if the SC is established, otherwise it is blocked. That methodology does not take into account the time required to set-up a VNF.

Based on MEC-First, we have developed a version of the method which only solves the VNF-Chaining problem. Therefore, the modified MEC-First dynamically solves the SC for each request, but it only employs the existing VNF instances that the planning method computed and allocated in the AOs and CO in the first stage, and it is not able to set up new ones at any node of the network. Consequently, when there are not enough VNF instances or network bandwidth to build the SC, the service is blocked.

Moreover, inspired in MEC-First, we also propose another method called CO-First, which operates in a similar manner: The algorithm starts the chaining process at the CO, first looking for existing instances of the required VNFs to establish the SC associated to the demanded traffic type. If there are not available instances, but there are enough IT resources at the CO, the algorithm sets up the necessary instances. If the chaining process cannot continue at the CO, due to lack of resources and there is enough available bandwidth between the CO and the local AO serving the end user, the chaining process continues at the local AO in the same manner, i.e., first employing existing instances of the required VNFs, then creating them if possible. Once in the AO, the algorithm is not able to use instances of the CO again. As in the previous method, if the SC is established then the traffic demand is served, otherwise is blocked. This method will be mainly used for performance comparison purposes.

In this paper, we use the modified MEC-First in order to solve the VNF-Chaining subproblem and, therefore, no VNF instances are created during the operation. The following subsections present the proposed method to solve the first stage, i.e., the VNF-Placement (or planning) method.

3.4. Genetic Algorithm for Service Mapping

Genetic algorithms are metaheuristics based on the mechanics of natural selection and evolution, which are commonly used to solve search and optimization problems [35]. In this section, we present GASM (Genetic Algorithm for Service Mapping), a genetic algorithm that solves the VNF-Placement

problem by planning the type and number of VNF instances to be instantiated in the MEC resource of AOs and CO. The algorithm takes into account the method used to build the required SCs to satisfy all the service requests of the users. Thanks to its use, it is possible to minimize, as its primary objective, the service blocking rate, and, as its secondary objective, the number of active CPU cores (and hence, the energy consumption), while considering restrictions in both the IT and network resources.

The basic version of GASM, which only operates on static scenarios, was presented in Reference [14]. In GASM, each potential solution is considered as an individual and it is represented by a chromosome composed of genes. Each gene encodes the number of instances of a given VNF that each node must host. Figure 2 shows an example of a chromosome employed by GASM. In this example, the AO_1 would host one instance of the VNF_1 and would not host instances of the VNF_n, whereas AO_m would host two instances of VNF_1 and four of VNF_n and the CO would host five instances of both VNFs.

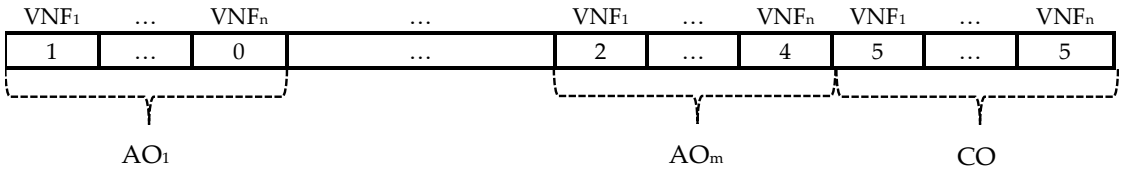


Figure 2. Example of a chromosome.

When translating the chromosome (i.e., converting the chromosome into a solution of the problem), GASM simulates the creation of the number of instances of each VNF at each location as described in the chromosome. Next, the algorithm sorts the user service requests, according to the operator preferred priority. After that, the algorithm establishes the SC for each request, i.e., checks the corresponding SC associated with the service and searches for the required VNFs to build the chain. For the chaining process, the algorithm employs the modified MEC-First method described in the previous section, which only uses the existing instances that were established in the previous stage. Therefore, the policy initially searches for the required VNFs at the MEC server hosted at the AO attending the request. If the local MEC server does not host an available instance of the VNF, the algorithm tries to find an available one at the CO. Once in the CO, the algorithm will not look for instances of the VNF back at any local MEC server. A service blocking occurs when the algorithm is unable to set up the SC, be it for lack of IT resources or bandwidth availability in the network.

For the evolution GASM uses a classical genetic algorithm loop (Algorithm 1) [35]. Firstly, it generates an initial population composed of a set of individuals randomly generated. Two ad-hoc individuals are also included to boost the performance of the algorithm. The first ad-hoc individual is the result of solving the VNF provisioning problem with the MEC-First policy as proposed in References [32,33], which was described in detail in the previous subsection. The second individual is the result of solving the VNF-Provisioning problem using the CO-First policy described in the previous subsection.

The next step executes classical genetic operators ("crossover" and "mutation") [35] in the population. The crossover (Algorithm 2) is done by randomly selecting two individuals from the population (parents), selecting an equal random position in both chromosomes, and then dividing those chromosomes into two parts. The offspring is produced by interchanging the second part of the parent chromosomes. The mutation (Algorithm 3) is applied to each gene with a probability: *mutationProbability*. When the gene has to be mutated, a new value of that gene is generated using a random uniform distribution among the possible values of the gene. The resulting individual undergoes a validation procedure in which the algorithm tries to translate the solution, emulating the creation of the number of VNF instances in every host as its chromosome specifies. If the individual created is not valid, i.e., if the algorithm cannot create the indicated instances of each VNF in every host, due to lack of IT or network resources, it discards the chromosome and creates a new individual

using the genetic operators (i.e., crossover and mutation). The algorithm repeats the process until achieving a desired population size. The last step is the calculation of two fitness parameters for each individual: The service blocking ratio and the percentage of required active CPU cores. The algorithm then selects the individuals with the best fitness performances to be the parent population of the following generation: GASM selects as preferred those individuals with a lower service blocking ratio and, to resolve the ties, it uses the number of actives CPU cores. The algorithm repeats this operation for as many generations as desired.

Algorithm 1: GASM

```
1:  procedure GASM(trafficEstimation, populationSize, nextPopulationSize, numberOfGenerations)
2:      solution ← Ø
3:      population ← Ø
4:      population ← generateAdHocIndividuals
5:      while size(population) < populationSize do
6:          population ← population ∪ generateRandomIndividual
7:          discardNonFeasibleSolutions(population)
8:      end while
9:      i ← 0
10:     while i < numberOfGenerations do
11:         nextPopulation ← Ø
12:         while size(nextPopulation) < nextPopulationSize do
13:             nextPopulation ← nextPopulation ∪ crossover(population)
14:         end while
15:         nextPopulation ← mutation(nextPopulation)
16:         discardNonFeasibleSolutions(nextPopulation)
17:         nextPopulation ← population ∪ nextPopulation
18:         nextPopulationIndividualFitness ← fitnessEvaluation(nextPopulation)
            population, populationIndividualFitness ← selectFittestIndividuals(nextPopulation,
19:                                                     nextPopulationIndividualFitness,
                                                          size = populationSize)
20:         i ← i + 1
21:     end while
22:     solution ← selectFittestIndividuals(population, populationIndividualFitness, size = 1)
23:     if solution ≠ currentlyEstablishedSolution then
24:         establishVNF(solution)
25:     end if
26: end procedure
```

At the end of the procedure, the algorithm returns the individual presenting the best performance in terms of service blocking ratio, or the solution with minor CPU core usage, if any tie appears. Hence, the result will represent the configuration of the NFV-enabled 5G network.

Algorithm 2: Crossover

```
1:  procedure crossover(population)
2:      parentA ← random(population)
3:      parentB ← random(population)
4:      if parentA ≠ parentB then
5:          randomPoint ← random(0, size(individualA))
6:          newIndividual[0, randomPoint −1] ← parentA[0, randomPoint −1]
7:          newIndividual[randomPoint, size(parentB)-1] ← parentB[randomPoint, size(individual)-1]
8:      end if
9:      return result
10: end procedure
```

Algorithm 3: Mutation

```
1:  procedure mutation(population, maxVNFPerLocation, mutationProbability)
2:    for individual in population do
3:      i ← 0
4:      while i < size(individual) do
5:        if random(0, 1) < mutationProbability then
6:          individual[gene] ← random(0, maxVNFPerLocation)
7:        end if
8:        i ← i + 1
9:      end while
10:   end for
11:   return population
12: end procedure
```

3.5. Reconfiguring VNF Provisioning with GASM

GASM provides a VNF-enabled network configuration well fitted to solve the service requests coming from a certain number of users, estimated as an average value. However, this kind of network design is not adapted to the time-varying traffic, therefore causing inefficient usage of IT resources. Consequently, a rise of the service blocking ratio may appear when the number of users is higher than the average. However, a decrease in the number of users may cause the over-consumption of IT resources, leading to a higher energy consumption.

In order to increase the performance of the network in a dynamic scenario, the best alternative is to reconfigure the VNF mapping. When a reconfiguring scenario is considered, the time is divided into time slots and the VNF provisioning algorithms are launched at the beginning of each slot. In order to predict the traffic in the following slot, we have considered a very simple method, shown in Equation (1):

$$T_{j+1} = T_j + \alpha\left(T_j - T_{j-1}\right), \tag{1}$$

where T_{j+1} represents the number of users at the next time slot (i.e., the estimation), T_j represents the current slot total number of users and T_{j-1} represents the total number of users at the previous temporal slot. The scale factor α is the maximum variation in the number of users (measured with traffic monitors) from one time slot to the next one, typically considering only a window of time, i.e., a day or a week. Despite its simplicity, this basic estimation method provides satisfactory results as will be shown in the following section. Once the traffic for the following time slot is estimated, GASM uses that information to determine a VNF provisioning and establish the new mapping in the IT resources of the MEC and CO.

3.6. Evolutive GASM

The performance of GASM in reconfiguration scenarios can be improved by including a simple learning technique. We will denote this new version of GASM by Evolutive GASM.

Since the simulation time is divided into discrete time slots, if the granularity of the step interval is sufficiently small, it is possible to assume that the traffic will not drastically change from one time slot to the next and, in consequence, it could be expected that the configuration for the current time slot will also present an adequate performance in the next. Therefore, Evolutive GASM utilizes the current provisioning configuration as one of the individuals of the initial population together with the ad-hoc individuals and the randomly generated ones, until completing the desired population size. The addition of this individual is expected to facilitate the search of a solution with better performance in terms of services blocking ratio or CPU usage than the resulting individual that GASM returns.

4. Performance Evaluation of GASM Algorithms over Cloud/MEC RAN Architecture

4.1. Simulation Settings

We have analyzed the performance of GASM in two scenarios: Planning a static distribution of VNF provisioning in a 5G access network, and a dynamic scenario with a periodical reconfiguration of that VNF provisioning. We have also compared these two alternatives between them. The analysis has been performed with the OMNeT++ simulator and supposing a network architecture similar to the one in Figure 1 with one CO connected to 20 AOs: 10 with a high number of served users (HD-AO) and other 10 with a low number of served users (LD-AO). We consider the AOs to be NFV-enabled, i.e., they are equipped with a MEC server to host both the vBBUs of the C-RAN architecture and also instances of different VNFs to provide end-user services. The AOs connect to the CO through 10 Gbps dedicated optical links. Note that for survivability, this architecture can also be embedded in an optical ring using Wavelength Division Multiplexing (WDM) or Space Division Multiplexing (SDM) technologies. Furthermore, the CO is also equipped with IT resources to instantiate VNF, but it will not host vBBUs. In our simulations, the IT resource distribution in CO, LD-AO and HD-AO (after the instantiation of vBBUs) are shown in Table 1. These resources are equal to those presented in papers [13,23].

Table 1. IT resource distribution in CO and AOs [13,23].

Location	Computational Resources
Core Office	100 CPU cores, 480 GB RAM and 27 TB HDD
High Demand—Access Office	16 CPU cores, 64 GB RAM and 10 TB HDD
Low Demand—Access Office	8 CPU cores, 32 GB RAM and 7 TB HDD

In the study, we have considered three types of services: VoIP, Video and Web, each with an associated SC and service bandwidth requirements, shown in Table 2. Additionally, each instance of a VNF has associated IT requirements in terms of CPU Cores, RAM and Hard Disk (HDD) and a maximum number of concurrent operations, which are shown in Table 3. Users may request these services with a probability of 30%, 20%, and 50% respectively, like in References [32,33]. The service priority established in the algorithm is first the VoIP service requests, then, the Video demands, and lastly the Web browsing demands.

Table 2. Requirements of the deployed service chains [13,23].

Service	Chained VNFs *	Bandwidth
VoIP	NAT-FW-TM-FW-NAT	64 Kbps
Video	NAT-FW-TM-VOC-IDPS	4 Mbps
Web Services	NAT-FW-TM-WOC-IDPS	100 Kbps

* NAT: Network Address Translator, FW: Firewall, TM: Traffic Monitor, WOC: WAN Optimization Controller, VOC: Video Optimization Controller, IDPS: Intrusion Detection Prevention System.

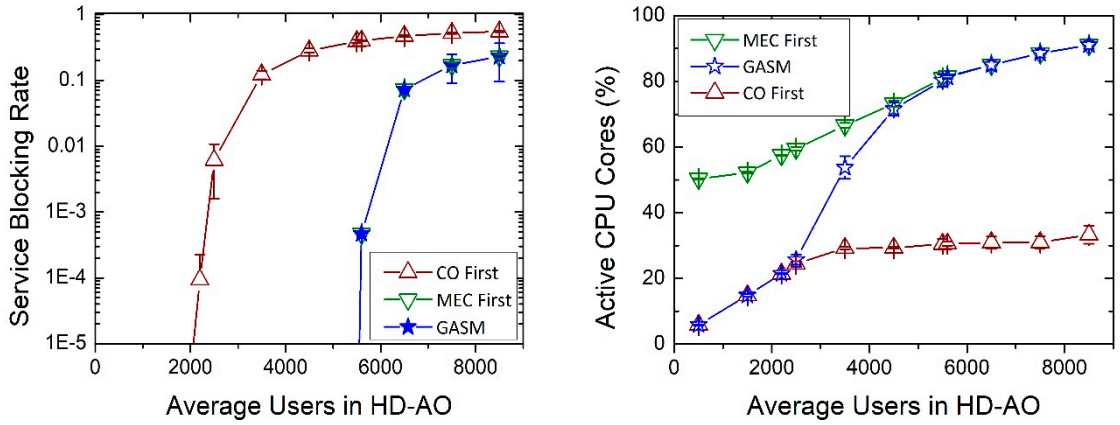
Table 3. Hardware requirements associated to the VNFs [13,23,32,33].

Service	HW Requirements.	Number of Concurrent Operations
NAT	CPU: 1 core, RAM: 1 GB, HDD: 2 GB	3000
FW	CPU: 2 cores, RAM: 3 GB, HDD: 5 GB	2500
TM	CPU: 1 core, RAM: 3 GB, HDD: 2 GB	2500
VOC	CPU: 2 cores, RAM: 2 GB, HDD: 20 GB	1000
WOC	CPU: 1 core, RAM: 2 GB, HDD: 10 GB	1500
IDPS	CPU: 2 cores, RAM: 2 GB, HDD: 10 GB	2500

Regarding GASM configuration, we have set the *populationSize* to 5 and the *nextPopulationSize* to 10 and the *mutationProbability* to 0.01. After performing simulations with different values of these parameters, we adopted these values as GASM provides the best results with them.

4.2. Planning with GASM in a Non-Reconfigurable Scenario

Firstly, we analyze the performance of GASM in a non-reconfigurable scenario and compare it with other similar proposals. In this scenario, the algorithm receives the number of users to be served and the services that they require, and the planning algorithm performs the VNF provisioning and the construction of the SC for each service of each user. Therefore, the traffic is generated only considering an average number of users per AO using a random variable with uniform distribution. We tested diverse numbers of average users in HD-AO, from 500 to 8500 with an increment of 1000 users, and for each number of users, 100 simulations were run. All the results presented in the paper are plotted with the 95% confidence interval. Figure 3a shows the service blocking ratio, i.e., the probability of not establishing a requested service, due to the lack of network or IT resources, while Figure 3b shows the percentage of active CPU in both CO and AOs. GASM was configured to stop after creating 100 generations and its performance is compared to the MEC-First and CO-First policies [32,33].



(**a**) (**b**)

Figure 3. (**a**) Service Blocking Rate and (**b**) Percentage of CPU cores in use for Genetic Algorithm for Service Mapping (GASM), MEC-First and CO-First in a static scenario.

Results in Figure 3a show that GASM achieves the same service blocking rate as MEC-First, but GASM provides planning solutions that utilize a lower number of CPU cores (more than 80% of reduction for a low number of users) and, therefore, less energy consumption (see Figure 3b). When compared with CO-First, Figure 3b shows that GASM provides planning solutions that use the same number of resources than CO-First for an average number of users per HD-AO up to 3500, but GASM uses more computing resources for higher traffic loads. However, taking into account the service blocking ratio, the one obtained with GASM is much lower than that of CO-First: E.g., if we consider that 10^{-3} is an acceptable value of service blocking ratio, the planning designed by GASM can deal with an increment over 200% in the number of users when compared with that of CO-First. Therefore, we can conclude that GASM outperforms MEC-First and CO-First in the achieved balance between service blocking rate and the CPU core usage [14].

Nevertheless, this traffic scenario cannot be considered realistic since the number of users requiring services from the network varies with time in real life. Hence, the behavior of GASM

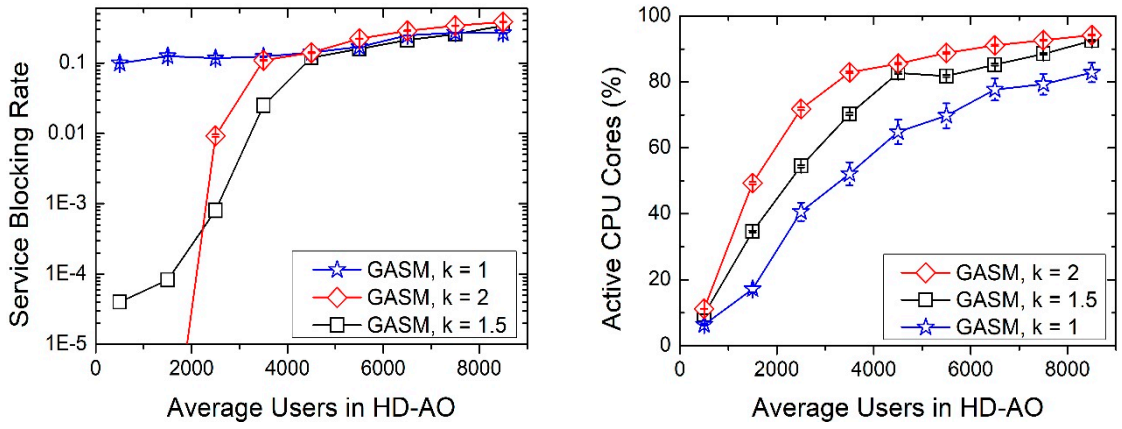
was also tested in a non-reconfigurable, but dynamic, scenario in which the number of users requiring services from the network varies following Equation (2) [36],

$$users_i(t) = \overline{users}_i \beta(t) \left[1 + \phi \sin\left(\frac{2\pi t}{t_{variation}}\right)\right], \quad (2)$$

where $\overline{users}_i$ represents the average number of users in the AO i and is a random variable generated using a uniform distribution $U\left[0,\ 2{\cdot}\overline{HDusers}\right]$ for HD-AO and U$[0,\ 2{\cdot}\overline{HDusers}/10]$ for LD-AO. In the study, values of $\overline{HDusers}$ range from 500 to 8500 users. ϕ represents the traffic variability and $t_{variation}$ represents the variation period in seconds. We chose $t_{variation} = 86,400\ s$, i.e., one day. The equation also includes $\beta(t)$ to incorporate the bursty nature of the traffic. It is generated every time that $\overline{users}_i(t)$ is evaluated and it is modeled as a random variable following a uniform distribution $U[1-\epsilon,\ 1+\epsilon]$, where ϵ represents the level of traffic burstiness and it is set to 0.1 (i.e., 10%) in our simulations.

An alternative to devise this dynamic scenario would be to plan the VNF provisioning considering the average number of users during a day in each AO (which can be obtained through the use and analysis of traffic monitors) and using a scaling factor to over-dimension the planning and serve the users even in the rush hours. Therefore, the number of users in an AO that the planning algorithm employs to design the VNF mapping is $k\Lambda_i$ where Λ_i is the average number of users in each AO and k is a scaling factor. The problem with the static planning is that GASM designs a provisioning which is not totally adapted to a scenario with time-varying traffic. Consequently, the provided VNF provisioning may lead to over use of computing resources and, consequently, to an increase of both the energy consumption and OPEX, as well as to a possibly higher service blocking rate.

Figure 4a shows the service blocking ratio with three different scaling factors: $k = 1$, $k = 1.5$ and $k = 2$. For each number of users we repeated the simulation 300 times (i.e., 300 values of Λ_i in Equation (1) for the number of users connected to each AO). Then, at the beginning of each simulation, we assume that the average number of users in each AO, Λ_i, has been estimated, and the VNF provisioning is done at the start of the simulation and not reconfigured during the three-day simulation. The corresponding percentages of CPU cores in use are shown in Figure 4b.



(**a**) (**b**)

Figure 4. Network planning results using GASM with k = 1, k = 1.5 and k = 2: (**a**) Service blocking rate, (**b**) percentage of CPU cores in use.

Figure 4a shows that the best results in terms of service blocking ratio in the simulated scenario are obtained when k = 1.5. With $k = 1$, the number of users is under-estimated and, therefore, the network and the VNF provisioning configuration are not prepared to deal with that number of users. Consequently, the service blocking ratio is very high, so that it cannot be used in a real network. Contrarily, if k = 2, the number of users in the network is over dimensioned leading to an

unnecessary increment in the number of CPU cores in use, and even the service blocking ratio is higher than that obtained with k = 1.5. Hence, it is possible to conclude that making a static planning is very complex as it is necessary to scale the estimated traffic in a proper manner, which is a difficult task. Using a low scaling factor would lead to an increment of the service blocking ratio, whereas an over-dimensioning of the network would lead to an unnecessary increment of the number of the CPU cores in use.

Finally, the execution time that GASM required to create the VNF mapping is shown in Figure 5. The results show that GASM requires more time to find a solution as the number of average users increases but it is always less than 240 s in a machine equipped with AMD Opteron 6128 and 64 GB RAM.

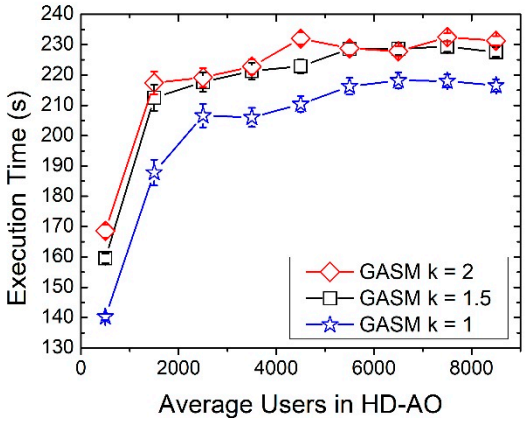


Figure 5. Execution times for GASM with k = 1, k = 1.5 and k = 2.

Previous results show the advantage of using GASM for solving the VNF provisioning, but they also show how inefficient it is to plan the VNF provisioning configuration in a static way. Consequently, we propose VNF reconfiguration using GASM with the expectation of improving both the service blocking probability and CPU usage.

4.3. Using GASM in a Reconfigurable Scenario

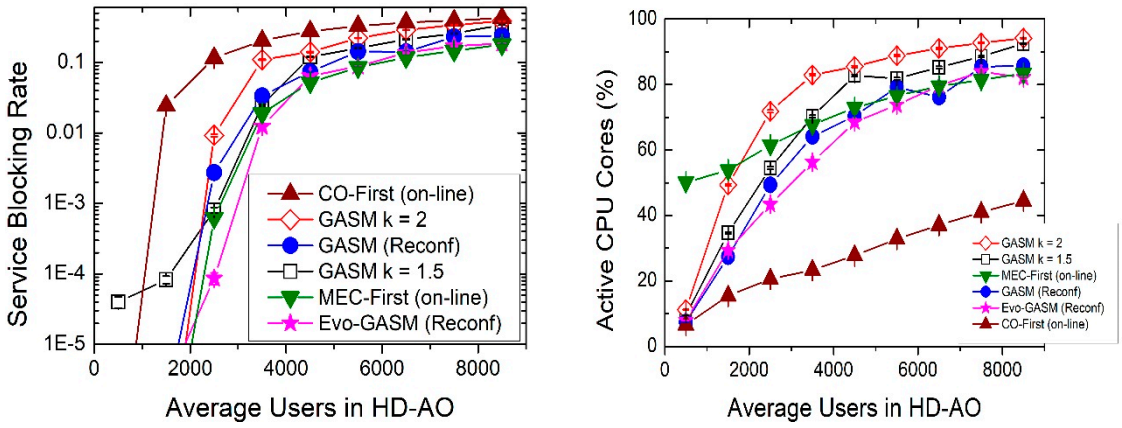
When GASM or Evolutive GASM is used to reconfigure the VNF provisioning, the time is divided into time slots and GASM is launched at the beginning of each slot. The duration of the time slot is set to 900 s as it is the one that offers the best results in terms of service blocking rate and CPU usage (not shown due to the lack of space).

In the following studies, we compare the performance of static planning (with k = 1.5 and k = 2) with the provisioning obtained using GASM and Evolutive GASM when a reconfiguration is allowed. In the static planning, GASM method evolves during 100 generations as the planning is done off-line and no time restrictions are required, but with reconfiguration methods, only 10 generations are created to cut down the execution time. Note that the computing time must be lower than the duration of a time slot. MEC-First and CO-First are launched online, therefore, they work with real traffic and they add/delete VNF as required similarly as it is done in References [32,33].

Figure 6a shows the global service blocking ratio depending on the average number of users in each HD-AO. Figure 6b shows the corresponding percentage of CPU core saving respect to static planning with k = 2. The corresponding execution times are shown in Figure 7.

Figure 6a shows that making a static planning leads to a reduction in the network performance. GASM $k = 2$ is the algorithm with the highest percentage of CPU usage and its service blocking rate is one of the highest. This happens as the VNF planning is over-dimensioned and it is not properly designed. When $k = 1.5$ better results can be obtained, but it is worse than those obtained by basic GASM with reconfiguration. On the other hand, CO-First (which does not exploit edge computing capabilities) presents the highest values of service blocking rates. Hence, not only does the use of MEC

help to reduce the delay, but it also improves the performance of the network. Although MEC-First obtains good results in terms of service blocking rate, Figure 6b shows that the algorithm makes a higher use of CPU cores in comparison to other methods, particularly for the lower number of average users per AO.



(**a**) (**b**)

Figure 6. (**a**) Service blocking rate and (**b**) Percentage of the CPU Cores in use for the proposed methods. The results obtained when reconfiguration and online is performed are indicated with full symbols.

Additionally, the evolutive method obtains the best results in terms of service blocking rate: Almost an order of magnitude when compared to MEC-First and the static version with the optimal configuration (k = 1.5). Furthermore, when compared to basic GASM when reconfiguration is allowed, the service blocking ratio is reduced in almost two orders of magnitude, showing the effectiveness of the learning step added to the algorithm. Evolutive GASM also presents a good performance in terms of IT resources, outperforming MEC-First and GASM for k = 1.5 and reducing the percentage of CPU cores in use almost in 10%.

In terms of execution times (Figure 7), MEC-First and CO-First are capable of solving all the service requests in less than 1 s while GASM methods require more time to perform the planning (around 20 s more time). In any case, the time required is much lower than the time slot duration of 900 s and can be used with the time slot used in the study.

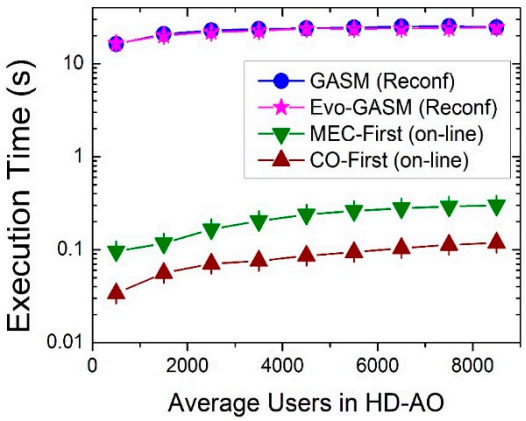


Figure 7. Execution times for basic GASM, evolutive GASM, MEC First and CO First for $\phi = 0.5$.

5. Conclusions

In this paper we address the VNF-Provisioning problem in a 5G access network with an architecture based on Cloud/MEC NFV-enabled RAN. We propose a novel genetic algorithm, GASM, to tackle the aforementioned problem, reducing both the service blocking rate and CPU usage (and, therefore, the energy consumption) considering limitations in IT and optical network resources. We address the problem from two different points of view: Static planning (where the provisioning is designed on a static traffic estimation) or a reconfiguration scenario in which the VNFs are reconfigured online depending on the current number of user service requests (changing dynamically) to be served. Moreover, we also propose a new version of GASM for the reconfiguration scenario which is enhanced with a learning technique causing an improvement in the performance of the method when reconfiguration is allowed.

By means of a simulation study, we show that our methods which outperform other proposals from the literature in both scenarios, achieving a favorable trade-off between the service blocking rate and the number of CPU cores actives. Results also show that reconfiguring the VNF-Provisioning reduces the service blocking rate and makes a more efficient use of the computation resources in the MEC and CO servers. Finally, our method enhancement incorporating the learning stage improves the service blocking rate performances by almost an order of magnitude respect to the other algorithms, producing the best balance between that metric and an efficient usage of IT resources.

Author Contributions: Conceptualization, all authors; Methodology, all authors; Software, L.R.; Investigation, L.R., R.J.D., I.M., J.-J.P.-M., P.P.-M., and P.S.K.; Visualization, L.R.; Writing-Original Draft Preparation, L.R., R.J.D. and I.M.; Writing-Review and Editing, all authors; Supervision, R.J.D., P.P.-M., and P.S.K.; Funding Acquisition, R.J.D., I.M., P.S.K., P.P.-M.

Funding: This work has been supported by Spanish Ministry of Economy and Competitiveness (TEC2014-53071-C3-2-P, TEC2017-84423-C3-1-P, TEC2015-71932-REDT), the fellowship program of the Spanish Ministry of Education, Culture and Sports (BES-2015-074514 and FPU14/04227) and European H2020-ICT-2016-2 METRO-HAUL (no. 761727).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. 5G Infrastructure Association. The 5G Infrastructure Public Private Partnership: the next generation of communication networks and services. 2015. Available online: https://5g-ppp.eu/ (accessed on 18 June 2018).
2. Chen, K.; Duan, R. *C-RAN the Road towards Green RAN*; China Mobile Research Institute: Beijing, China, 2011; Volume 2.
3. Pfeiffer, T. Next Generation Mobile Fronthaul Architectures. In Proceedings of the 2015 Optical Fiber Communications Conference and Exhibition (OFC), Los Angeles, CA, USA, 22–26 March 2015; OSA: Los Angeles, CA, USA, 2015; pp. 1–3.
4. CPRI Specification V7.0. p. 128. Available online: http://www.cpri.info/ (accessed on 28 November 2018).
5. De la Oliva, A.; Hernandez, J.A.; Larrabeiti, D.; Azcorra, A. An overview of the CPRI specification and its application to C-RAN-based LTE scenarios. *IEEE Commun. Mag.* **2016**, *54*, 152–159. [CrossRef]
6. Musumeci, F.; Bellanzon, C.; Carapellese, N.; Tornatore, M.; Pattavina, A.; Gosselin, S. Optimal BBU Placement for 5G C-RAN Deployment Over WDM Aggregation Networks. *J. Lightw. Technol.* **2016**, *34*, 1963–1970. [CrossRef]
7. 5G PPP Architecture Working Group View on 5G Architecture. White Paper, July 2016. Available online: https://5g-ppp.eu/ (accessed on 28 November 2018).
8. 5G-Oriented Optical Transport Network Solution. ZTE White Paper. Available online: http://www.zte.com._cn/global/solutions/network/ (accessed on 28 November 2018).
9. Eramo, V.; Listanti, M.; Lavacca, F.; Iovanna, P. Dimensioning Models of Optical WDM Rings in Xhaul Access Architectures for the Transport of Ethernet/CPRI Traffic. *Appl. Sci.* **2018**, *8*, 612. [CrossRef]
10. Patel, M.; Naughton, B.; Chan, C.; Sprecher, N.; Abeta, S.; Neal, A. *Mobile-Edge Computing Introductory Technical White Paper*; Mobile-Edge Computing (MEC) Industry Initiative; ETSI: Sophia Antipolis, France, 2014.
11. Peng, M.; Li, Y.; Jiang, J.; Li, J.; Wang, C. Heterogeneous cloud radio access networks: A new perspective for enhancing spectral and energy efficiencies. *IEEE Wirel. Commun.* **2014**, *21*, 126–135. [CrossRef]

12. Tinini, R.I.; Reis, L.C.M.; Batista, D.M.; Figueiredo, G.B.; Tornatore, M.; Mukherjee, B. Optimal Placement of Virtualized BBU Processing in Hybrid Cloud-Fog RAN over TWDM-PON. In Proceedings of the 2017 IEEE Global Communications Conference (GLOBECOM), Singapore, 4–8 December 2017; pp. 1–6.
13. Savi, M.; Tornatore, M.; Verticale, G. Impact of processing costs on service chain placement in network functions virtualization. In Proceedings of the 2015 IEEE Conference on Network Function Virtualization and Software Defined Network (NFV-SDN), San Francisco, CA, USA, 18–21 November 2015; IEEE: San Francisco, CA, USA, 2015; pp. 191–197.
14. Ruiz, L.; Durán, R.J.; Aguado, J.C.; Pavon-Marino, P.; Siddiqui, S.; Mata, J.; Fernández, P.; Lorenzo, R.M.; Abril, E.J. Genetic Algorithm for Effective Service Mapping in the Optical Backhaul of 5G Networks. In Proceedings of the 2018 20th International Conference of Transparent Optical Networks (ICTON), Bucharest, Romania, 1–5 July 2018; IEEE: Bucharest, Romania, 2018; pp. 1–4.
15. Kani, J.; Kuwano, S.; Terada, J. Options for future mobile backhaul and fronthaul. *Opt. Fiber Technol.* **2015**, *26*, 42–49. [CrossRef]
16. Rimal, B.P.; Van, D.P.; Maier, M. Mobile Edge Computing Empowered Fiber-Wireless Access Networks in the 5G Era. *IEEE Commun. Mag.* **2017**, *55*, 192–200. [CrossRef]
17. Wang, K.; Yang, K. Power-Minimization Computing Resource Allocation in Mobile Cloud-Radio Access Network. In Proceedings of the 2016 IEEE International Conference on Computer and Information Technology (CIT), Nadi, Fiji, 8–10 December 2016; IEEE: Nadi, Fiji, 2016; pp. 667–672.
18. Wang, X.; Wang, K.; Wu, S.; Sheng, D.; Jin, H.; Yang, K.; Ou, S. Dynamic Resource Scheduling in Mobile Edge Cloud with Cloud Radio Access Network. *IEEE Trans. Parallel Distrib. Syst.* **2018**, *29*, 2429–2445. [CrossRef]
19. Wang, W.; Zhao, Y.; Tornatore, M.; Li, H.; Zhang, J.; Mukherjee, B. Coordinating Multi-access Edge Computing with Mobile Fronthaul for Optimizing 5G End-to-End Latency. In Proceedings of the 2018 Optical Fiber Communication Conference and Exposition (OFC), San Diego, CA, USA, 11–15 March 2018; OSA: San Diego, CA, USA, 2018; pp. 1–3.
20. Blanco, B.; Fajardo, J.O.; Giannoulakis, I.; Kafetzakis, E.; Peng, S.; Pérez-Romero, J.; Trajkovska, I.; Khodashenas, P.S.; Goratti, L.; Paolino, M.; et al. Technology pillars in the architecture of future 5G mobile networks: NFV, MEC and SDN. *Comput. Stand. Interfaces* **2017**, *54*, 216–228. [CrossRef]
21. Lin, T.; Zhou, Z.; Tornatore, M.; Mukherjee, B. Optimal network function virtualization realizing end-to-end requests. In Proceedings of the 2015 IEEE Global Communications Conference (GLOBECOM), San Diego, CA, USA, 6–10 December 2015; IEEE: San Diego, CA, USA, 2015; pp. 1–6.
22. Gil Herrera, J.; Botero, J.F. Resource Allocation in NFV: A Comprehensive Survey. *IEEE Trans. Netw. Serv. Manag.* **2016**, *13*, 518–532. [CrossRef]
23. Savi, M.; Hmaity, A.; Verticale, G.; Höst, S.; Tornatore, M. To distribute or not to distribute? Impact of latency on Virtual Network Function distribution at the edge of FMC networks. In Proceedings of the 2016 18th International Conference on Transparent Optical Networks (ICTON), Trento, Italy, 10–14 July 2016; pp. 1–4.
24. Carpio, M.F.; Dhahri, S.; Jukan, A. VNF Placement with Replication for Load Balancing in NFV Networks. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017; IEEE: Paris, France, 2017; pp. 1–6.
25. Zeng, M.; Fang, W.; Zhu, Z. Orchestrating Tree-Type VNF Forwarding Graphs in Inter-DC Elastic Optical Networks. *J. Lightw. Technol.* **2016**, *34*, 3330–3341. [CrossRef]
26. Jia, Y.; Wu, C.; Li, Z.; Le, F.; Liu, A. Online Scaling of NFV Service Chains Across Geo-Distributed Datacenters. *IEEE/ACM Trans. Netw.* **2018**, *26*, 699–710. [CrossRef]
27. Wang, X.; Wu, C.; Le, F.; Liu, A.; Li, Z.; Lau, F. Online VNF Scaling in Datacenters. In *2016 IEEE 9th International Conference on Cloud Computing (CLOUD)*; IEEE: San Francisco, CA, USA, 2016; pp. 140–147.
28. Wang, X.; Wu, C.; Le, F.; Lau, F.C.M. Online Learning-Assisted VNF Service Chain Scaling with Network Uncertainties. In Proceedings of the 2017 IEEE 10th International Conference on Cloud Computing (CLOUD), Honolulu, CA, USA, 25–30 June 2017; IEEE: Honolulu, CA, USA, 2017; pp. 205–213.
29. Rankothge, W.; Le, F.; Russo, A.; Lobo, J. Optimizing Resource Allocation for Virtualized Network Functions in a Cloud Center Using Genetic Algorithms. *IEEE Trans. Netw. Serv. Manag.* **2017**, *14*, 343–356. [CrossRef]
30. Otokura, M.; Leibnitz, K.; Koizumi, Y.; Kominami, D.; Shimokawa, T.; Murata, M. Application of evolutionary mechanism to dynamic Virtual Network Function Placement. In Proceedings of the 2016 IEEE 24th International Conference on Network Protocols (ICNP), Singapore, 8–11 November 2016; IEEE: Singapore, 2016; pp. 1–6.

31. Nguyen, T.-M.; Fdida, S.; Pham, T.-M. A comprehensive resource management and placement for network function virtualization. In Proceedings of the 2017 IEEE Conference on Network Softwarization (NetSoft), Bologna, Italy, 3–7 July 2017; IEEE: Bologna, Italy, 2017; pp. 1–9.
32. Pedreno-Manresa, J.-J.; Khodashenas, P.S.; Siddiqui, M.S.; Pavon-Marino, P. Dynamic QoS/QoE assurance in realistic NFV-enabled 5G Access Networks. In Proceedings of the 2017 19th International Conference on Transparent Optical Networks (ICTON), Girona, Spain, 2–6 July 2017; IEEE: Girona, Spain, 2017; pp. 1–4.
33. Pedreno-Manresa, J.-J.; Khodashenas, P.S.; Siddiqui, M.S.; Pavon-Marino, P. On the Need of Joint Bandwidth and NFV Resource Orchestration: A Realistic 5G Access Network Use Case. *IEEE Commun. Lett.* **2018**, *22*, 145–148. [CrossRef]
34. Juarez, F.; Ejarque, J.; Badia, R.M. Dynamic energy-aware scheduling for parallel task-based application in cloud computing. *Future Gener. Comput. Syst.* **2018**, *78*, 257–271. [CrossRef]
35. Goldberg, D. *Genetic Algorithms in Optimization, Search and Machine Learning*; Addison-Wesley: Reading, MA, USA, 1989.
36. Gencata, A.; Mukherjee, B. Virtual-topology adaptation for WDM mesh networks under dynamic traffic. *IEEE/ACM Trans. Netw.* **2003**, *11*, 236–247. [CrossRef]