

Reliable Network Slices based on Elastic Network Resource Demand

Gabriel da Silva, Dyego Oliveira and Rafael L. Gomes
State University of Ceara (UECE)
Fortaleza – CE, Brazil
Email: {gabriel.galdino,doliveira,rafaellgom}@larc.es.uece.br

Luiz F. Bittencourt and Edmundo R. M. Madeira
University of Campinas (UNICAMP)
Campinas – SP, Brazil
Email: {bit,edmundo}@ic.unicamp.br

Abstract—Internet access is crucial to the human society as a platform for several services to the users. Despite this importance, the Internet suffers limitations that compromise Quality of Service (QoS) guarantees. Thus, Internet Service Providers (ISPs) need to evolve, adding new technologies and management strategies to their infrastructure. A promising approach is the slicing of network resources among clients and delivered services, where reliability and elastic resource demand through the day are key issues. Within this context, this paper presents an algorithm called Reliable Reuse Encourage (R-REENC), which defines network slices based on bandwidth requirements and the desired reliability for the clients. The results suggest that the proposed algorithm allocates more suitable slices than other existing approaches.

Index Terms—Slicing, Elasticity, Resource Demand, Internet Service Providers

I. INTRODUCTION

Internet Service Providers (ISPs) have a crucial role in the modern life, since Internet Access Service (IAS) is nowadays essential in the sense it supports applications in a range of areas, such as public services, security, industry, and entertainment. Due to this importance, ISPs extended their Internet access alternatives, providing different access networks (wireless/wired) for their clients (enterprises, small companies, home networks, third party service providers, directly to final users, etc) [1]. The access networks of an ISP are interconnected by a common edge network, which is the previous step before reaching the core of Internet [2].

Specific types of access network may present singular characteristics. However, regardless of their type, they deal with requirements such as low delay, Bandwidth (Bw) demands, reliability, and others. All these requirements have direct impact on the Quality of Service (QoS) and Quality of Experience (QoE) of the delivered IAS [3]. Based on this scenario, Network Virtualization (NV), Software Defined Network (SDN), and Network Function Virtualization (NFV) arose as promising technologies to overcome the existing limitations [4]. The simultaneous applications of these technologies enable the splitting of network resources into slices, where each network slice can have the most suitable configuration to better meet the client's requirements [5]. A network slice is a logical isolated network on the top of a physical (or virtual) network, being dynamically and independently managed [6]. Therefore,

a network slice is suitable to support a specific service delivery or even individual clients [1].

In general, Internet users become frustrated when they experience problems such as high delays, constant disconnections, and slowness in their applications. Thus, ISPs need to guarantee a certain quality to their IAS. The IAS is specified in a Service Level Agreement (SLA), defining the client requirements to be met by the ISP [3]. Reliability is a crucial requirement to be met by ISPs, since it affects the availability of the IAS. Reliability can be defined as the probability of the IAS to maintain the communication capacity when failures occur in the physical network infrastructure [7]. Reliability fulfillment is related not only to reactive actions to address post-failure impact, but also pre-failure strategic planning. Therefore, an ISP should deploy the IAS planning to mitigate a possible impact in the QoS/QoE of the clients in face of network failures.

An important point for ISPs to keep the quality of IAS is the elastic demand for network resources throughout the day, which occurs because of the human mobility within cities. In this scenario, ISPs need to expand or to shrink the bandwidth allocation dynamically (elastic behavior), allowing a suitable on demand IAS. A key task to provide elastic services is the slice allocation, i.e., the definition of which network elements (links and nodes) will compose each slice. Thus, the allocation task is part of the strategic planning to deploy the slices based on the identified demand. Furthermore, ISPs need to consider potential changes necessary in the slice structure to keep it suitable to meet the elastic demand, resulting in an adaptation in the slice structure (i.e., modification of network elements of the slice). The execution of an adaptation causes an impact under the network infrastructure performance [5].

Therefore, ISPs should take into account the best set of slices (each by the defined period of the day) to address the elastic demand. Nevertheless, this slice allocation task has no standard to deal with aspects of elastic demand and reliability simultaneously.

Within this context, this paper presents the Reliable Reuse Encourage (R-REENC) algorithm, which allocates network slices considering the Bw requested by the clients during the day and the minimum reliability required in the SLA. The goal of R-REENC is to define the most suitable configuration of network slices to decrease the network resource utilization

while it encourages the reuse of network elements from previous slices when a new demand is being processed (aiming to minimize the number of adaptations). In order to ensure the reliability requirements, the R-REENC algorithm applies a relative disjoint path strategy, searching for alternative paths until it reaches the desired reliability for the slice.

The performance of R-REENC was evaluated and compared with an existing approach to allocate network slices that performs the definition of network slices individually, called Per Time Calculation (PETIC) [8]. The results suggest that the proposed algorithm defines suitable network slices, enhancing the Bw usage, minimizing the number of adaptations and mitigating the impact of physical failures in the IAS quality.

The remainder of this paper is organized as follows. Section II describes the context of this paper, while Section III presents the related work found in the literature. Section IV introduces the proposed algorithm. Section V presents the results of the experiments performed. Finally, Section VI concludes the paper and presents future work.

II. CONTEXT DESCRIPTION

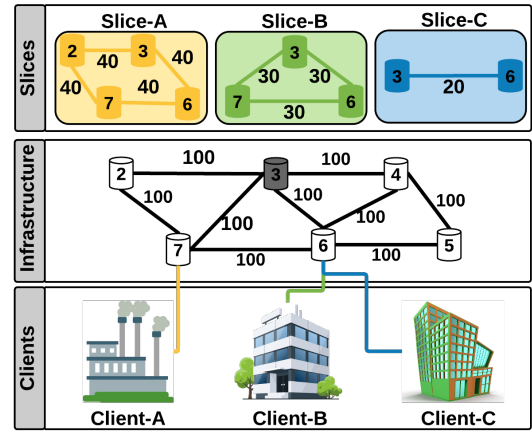
ISPs need to improve their service delivery to clients, since new network requirements arose in the last years due to the plethora of different applications that appeared recently. In this scenario, reliability, flexibility and isolation are key features to enhance QoS and QoE considering singularities in service requirements. One possible approach to address this scenario is the integration of SDN, NV, and NFV.

This integration creates customizable network slices from the network infrastructure, supporting and providing specific services. A network slice is linked to an SDN controller, which deploys the network applications via northbound protocols and communicates with NFV MANO to improve the deployment and connection of network services [6], [1]. Thus, each network slice can deploy network functions according to the characteristics described in the SLA. The slicing of the network infrastructure allows flexibility due to the capacity of the ISP (with a network hypervisor) to dynamically adjust the network slices characteristics, shaping it according to the current service demand.

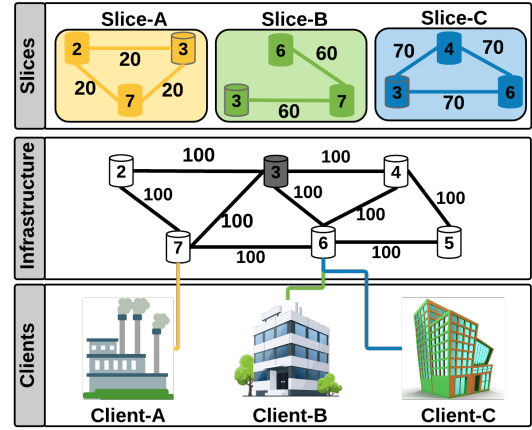
The configuration of a network slice contains some parameters that specify metrics to be considered in the allocation. The metric considered in this paper is the Bandwidth. It is important to notice that the context described is independent of the underlying transport network technologies used by the ISP, where the SDN technology and the network hypervisor are suitable approaches [9], [10].

Figure 1 shows an example of the described scenario, where three network slices are deployed to three different clients over the same network infrastructure (which has 100Mbps of available bandwidth in each link). Initially, each network slice is configured according to the characteristics of the demand and workload (Figure 1(a)), and later the slice is adapted to the new demand for the nighttime (Figure 1(b)).

In Figure 1, the ISP allocates *Slice-A*, *Slice-B* and *Slice-C* for *Client-A*, *Client-B* and *Client-C*, respectively. Each slice



(a) Resource demand during the day.



(b) Resource demand during the night.

Fig. 1. Example of Scenario

is configured based on the SLA deployed between clients and the ISP (one SLA per client), where the requirements of the service delivery are specified, including the bandwidth required by the client during the day. Therefore, the resource demand from clients changes through the day, characterizing an elastic demand of network resources.

In the example, *Client-A* negotiates an SLA with the ISP to interconnect nodes 7 and 3, and it deploys *Slice-A* (yellow) that has 40 Mbps during the day (Figure 1(a)) and 20 Mbps in the night (Figure 1(b)). In the same way, the ISP applies *Slice-B* (green) with 30 Mbps and 60 Mbps for the morning and night, respectively (as shown in Figure 1). Finally, *Slice-C* (blue) is modelled following the same idea, initially having 20 Mbps and later 70 Mbps. Thus, the ISP meets the resource demand for elastic bandwidth which occurs throughout the day according to the characteristics of the client.

An algorithm must be applied to define which components of the network will be part of each slice being deployed, following the requirements of the clients and the resource availability in the network infrastructure (such as available bandwidth). Besides the bandwidth, another important issue that should be considered is the adaptation of the network components (nodes and/or links) to deploy the slice. Adapta-

tions occur when a link does not have the requested bandwidth or it is considered not suitable (i.e, it compromises the network infrastructure performance). Therefore, the ISP needs to adjust the previously allocated slice to meet the new requirements for the currently period of time. This scenario is represented in Figure 1. In the beginning of the day (Figure 1(a)), Slices A, B, and C have an initial configuration for the respectively time of day. Later, the slices are adjusted, modifying their structure, since they do not have enough bandwidth.

The nighttime bandwidth demand of clients A, B, and C (20 Mbps, 60 Mbps and 70 Mbps, respectively) exceeds the available bandwidth necessary in the link between nodes 3 and 6 (100 Mbps), which is part of their slices. Therefore, the ISP adjusts the structure of the slices to meet the requirement of its clients for the period of the day. In the example, the ISP maintains part of the structure of slices, adding and removing a few links. The behavior of the algorithm used to decide how to adapt a slice impacts directly the number of adaptations, the bandwidth usage, and the energy consumption of the network infrastructure, since frequently allocating a link will quickly saturate it (fact that could compromise future slice allocations due to lack of links availables) and still affects the energy efficiency of the ISP.

Regarding reliability, this paper considers that, at any time and any place, single failure of a network elements happens randomly and independently from each other (following a probability previously known). Thus, each network elements can be in one of two states: operational or failed. The failure of a link means that it is removed from the network, while the failure of a node means that the node and all the links related to it are removed.

Based on this scenario, this paper presents an algorithm to perform network slices allocation, following the elastic bandwidth demand during the day considering reliability constraints. Additionally, the algorithm considers requirements of the client (bandwidth requested through the day), as well as the adaptation impact over the network infrastructure.

III. RELATED WORK

This section discusses a few crucial papers regarding resource management, network slicing, NV, NFV, and SDN. Table I summarizes the papers discussed, highlighting the difference between them and our proposal. Each column of Table I points out a feature of each paper: *Context* indicates the network environment of the papers, while *Focus* summarizes the problem addressed in the paper.

Son et al. [11] propose a mechanism to provide network resources for Virtual Network Functions (VNFs) deployment, adapting to dynamically changing network volumes. This proposal considers the latency requirement of different applications in the service function chain, enabling the latency-sensitive applications to minimize the end-to-end network delay. Nevertheless, this proposal is limited to resource provision for VNFs, disregarding reliability and adaptation issues.

Leconte et al. [12] present a framework to optimally distributed cloud infrastructures interconnected by SDN. This

framework allows the network resource allocation for slices both in terms of Bw and cloud processing. This framework includes an algorithm to reach the optimal resource allocation. However, this proposal does not consider the elastic demand of network resources throughout the day, as well as reliability issues.

Caballero et al. [13] describe the NETwork Slicing (NES) framework to maintain the system in a Nash equilibrium, considering resource allocation and a user dropping method (based on an admission control policy). The NES framework applies the admission control to guarantee that slices can satisfy the rate requirements of all their users. Nevertheless, NES does not consider a strategic planning approach to define the slices structures nor the adaptations necessary on them.

Kwak et al. [14] propose a wireless network virtualization framework for long time-scale dynamic resource allocations, encompassing the coexistence of different services in a single wireless network infrastructure. The authors differentiate two services by analyzing objectives and constraints to successfully allocate the resources. It focuses on scheduling and power control for IoT services and for video streaming services. Although this paper takes into account the time of day to slice the network, it does not consider reliability constraints and it is restricted to wireless networks.

Han et al. [15] define a system to guarantee Bw in an application-aware end-to-end (E2E) slices definition. Moreover, this system applies the idea of software-defined wireless and NFV. However, this paper is not concerned about reliability and different periods of the day, nor with the possible impacts of the strategic planning during the slices definition.

Wang et al. [16] formulated a Slice Provider's Problem (SPP) to maximize the resource efficiency focusing on the profit of service providers and a Slice Customer's Problem (SCP) to maximize the profit of individual service customers. Nevertheless, this paper does not take into consideration the time of the day when slicing the network and the reliability of the slices for the service provision.

TABLE I
RELATED WORK

Reference	Context	Focus
Son et al. [11]	NFV	Dynamic resource provision for VNFs
Leconte et al. [12]	SDN	Optimization framework for Clouds in SDN
Caballero et al. [13]	Wireless	Framework for Nash equilibrium
Kwak et al. [14]	Wireless	Dynamic resource allocation
Han et al. [15]	E2E	On-demand application-aware end-to-end slices
Wang et al. [16]	5G	Maximize profit and resource efficiency
This work	ISP	Reliability and elastic Bw

To the best of our knowledge, no work in the literature aims to develop an algorithm to allow ISPs to deal with the elastic bandwidth that occurs during the day, while reducing the number of slices adaptations and improving the reliability of the IAS. Therefore, the proposal of this paper evolves the existing approaches of network slice allocation to address the new requirements of the ISPs.

IV. PROPOSAL

This section details the proposed R-REENC algorithm, as well as all the basic information necessary to understand its design and execution. In this way, it was split into three subsections to describe the problem to be addressed, the operation of the proposed algorithm, and methods applied by R-REENC. Table II summarizes the notation applied through this section.

TABLE II
NOTATION

Symbol	Description
G	Graph representing the network infrastructure $G = (N, L)$
$\bar{G}$	Graph with part of the network elements in G
$N = \{n_1, \dots, n_k\}$	Set of k nodes.
$L = \{l_1, \dots, l_m\}$	Set of m links.
l	link between two nodes
$l_{w,t}$	cost / weight of link l in period of time t
$l_{Bw,t}$	available bandwidth Bw in the period of time t of link l
$Bw_{c,t}$	bandwidth required by the client c for the time slot t
C	set of c client's active SLAs with the ISP. Therefore, $C = \{c_1, \dots, c_c\}$
T	s allocation time periods defined in the SLA by the clients ($T = \{t_1, \dots, t_s\}$)
t	specific period of time in T
c	the client's edge point to the ISP
F	set of network slices that need to be allocated to attend the clients of the ISP
$F_{i,t}$	represents the network slice for client i in period t
$\bar{F}_{i,t}$	set of additional network elements to $F_{i,t}$
S_i	minimum reliability requested to the network slice of client i
$R_{i,t}$	reliability of network slice $F_{i,t}$
ε	constant close to zero
p	percentage of redundancy ($0 \leq p \leq 1$)
W	best cost during the search process

A. Reliable Reuse Encourage (R-REENC)

In this section the R-REENC algorithm is presented. It aims to define a network slice ($F_{i,t}$) for each period of time (t in T) according to the Bw demand ($Bw_{c,t}$) and the minimum reliability (S_c) requested by the client (c). The principle of R-REENC is to find the most suitable network slice structure, while it encourages the reuse of the previous network slice. An overview of R-REENC is shown in Algorithm 1.

Set T is used to model the periods of time in which the client wants to specify a Bw demand. These values can be fixed, predicted by the ISP, dynamically requested by the client, or a hybrid approach. Thus, the R-REENC algorithm only needs a Bw demand ($Bw_{c,t}$) to be allocated, regardless its origin. In the same way, R-REENC is capable to work with all possible sizes of T . This fact eases the client to model T according to his reality, i.e., T can model: each hour of the day, providing 24 values ($T = \{t_1, \dots, t_{24}\}$); several minutes, for example 48 values representing 30 minutes, etc.

Basically, R-REENC analyzes a request of the client c , passing through the T periods of time. The variable p is responsible for the redundancy control, i.e., determining how many links need to be updated. p is incremented in the end

of each iteration to analyze the possible configurations. To update a link means to assign a weight (l_w) of ε (a quantity close to zero) to avoid the use of a link in the next step of the allocation.

This situation can be exemplified in the following cases: (I) $p = 0$ means no redundancy, i.e., the generated network slice will have just one path to interconnect the nodes; (II) $p = 1$ is the full redundancy case, i.e., the network slice will have two fully disjoint paths to interconnect the nodes; (III) $p = 0.5$ is the case where half of the links in the primary path will also compose the secondary path.

Algorithm 1 presents the steps performed during R-REENC execution. Firstly, between lines 3 and 9, the weights for the links ($l_{w,t}$) for the period of time T are assigned. The function applied in line 7 aims to avoid the saturation of the Bw in the links, as well as enables the increase of the allocation in cases of higher Bw demand in the next period of time. This function was originally proposed in [8], where more details about its behavior are provided.

Regarding the suitability of links, if the previous network slice allocated can be used as a basis (i.e., the links have the Bw demand), the reuse of these links is encouraged by assigning a weight close to zero (ε) by function *UpdateSlice(Graph, NetworkSlice, Redundancy)*. This reuse approach decreases the number of adaptations necessary in the network slices structure and, consequently, the impact of these adaptations in the network infrastructure performance.

Line 10 executes the *Search(Graph, Request)* function to find the most suitable path in the network infrastructure. The details of *Search(Graph, Request)* function will be presented in Section IV-B. From this function, the R-REENC algorithm has two possible situations: (i) a base structure for $F_{c,t}$ is defined; or, (ii) the search for a suitable path failed, since the network infrastructure does not have the Bw demand (verified between lines 12 and 14).

When the network slice $F_{c,t}$ is successfully defined (i.e., it meets $B_{c,t}$), it is checked if $F_{c,t}$ reaches the minimum reliability for client c ($R_{c,t} \geq S_c$). If this is the case, $F_{c,t}$ is included in the set F (line 23), otherwise the reliability strategy is applied (from line 14 to 19). The reliability calculation method used is presented in Section IV-C.

The reliability strategy aims to add alternative relative disjoint paths regarding $F_{c,t}$, where the redundancy of $F_{c,t}$ is incrementally increased (controlled by variable p). In line 17, the definition of the alternative paths for the network slice is performed, creating $\bar{F}_{c,t}$. In the end, both $\bar{F}_{c,t}$ and $F_{c,t}$ are merged, verifying if this new network slice structure is capable of reaching the desired reliability (S_c).

In nutshell, the reliability strategy increases the connectivity alternatives of the network slice until it reaches the full redundancy ($p = 1$). This approach aims to limit the Bw consumed, since the allocation of many links in the network slice tends to saturate the network resources of the physical infrastructure.

Regarding the time complexity, the search process (detailed in Section IV-B) is $O(k \log k)$, the slice links update is

Algorithm 1 R-REENC

Require: Requests of client C and Network Infrastructure G **Ensure:** Set of Network Slices F

```
1: for all Request  $c \in C$  do
2:   for all Period  $t \in T$  do
3:     for all Link  $l \in L$  do
4:       if  $l \in F_{c,t-1}$  and  $(l_{Bw,t-1} \nmid B_{c,t})$  then
5:          $l_{w,t} = \varepsilon$ ;  $\triangleright l$  can be used in period  $t$ 
6:       else
7:          $l_{w,t} = \exp\left(\frac{l_{Bw,t}}{B_{w,t}}\right)$ ;
8:       end if
9:     end for
10:     $F_{c,t} = \text{Search}(G, c)$ ;
11:    if  $F_{c,t} == \text{Fail}$  then
12:      return Fail;  $\triangleright$  Bw for client  $c$  cannot be met
13:    else
14:      while  $R_{c,t} < S_c$  and  $p \leq 1$  do  $\triangleright$  Redundancy
15:         $p += 0.25$ ;
16:         $\overline{G} = \text{UpdateSlice}(G, F_{c,t}, p)$ ;
17:         $\overline{F}_{c,t} = \text{Search}(\overline{G}, c)$ ;
18:         $F_{c,t} += \overline{F}_{c,t}$ ;
19:      end while
20:      if  $R_{c,t} < S_c$  then
21:        return Fail;  $\triangleright$  Reliability for client  $c$ 
22:        cannot be met
23:      else
24:         $F_{c,t} = \text{Finclur}(F_{c,t})$ ;
25:      end if
26:    end for
27:  end for
28: return  $F$ ;  $\triangleright$  Set of Network Slices
```

$O(m)$ and the reliability calculation method (described in Section IV-C) is $O(k^2m + km \log m)$, where k and m are the number of nodes and links in the network infrastructure. Thus, Algorithm 1 has the complexity of $O(k^2m + km \log m)$, representing that the dominant part of the computation time is the reliability calculation. From this fact, it is worth to mention that the algorithm proposed in this paper is agnostic of the reliability calculation method, i.e., different reliability calculation methods can be utilized with the proposed algorithm.

B. Search Process

An important part of *R-REENC* algorithm is the search process (lines 10 and 17 of Algorithm 1), which is presented in Algorithm 2. This algorithm was originally proposed in reference [8] to find suitable paths inside a network topology according to link weights ($l_{w,t}$).

In the first steps, the set of nodes to be analyzed is defined, excluding the initial node (the client's edge point to the ISP). Next, from lines 2 to 8, the cost of the links are assigned, where W represents the cost to reach each node in the physical network. The neighbors of client c receive the cost

$l_{w,t}$ (computed by the function $\text{getLinkInfo}(\text{Node}, \text{Node})$), while ∞ (infinity) is assigned to the remaining neighbors. Between lines 9 and 22 the search process is performed, aiming to find the set of links that are most suitable to the requirements for the network slice.

Algorithm 2 Search Process

Require: Network Infrastructure G and Client Request c **Ensure:** Slice f or *Fail*

```
1:  $S = N - \{c\}$ ;
2: for all Node  $j \in S$  do
3:    $W_j = \infty$ ;
4:    $l = \text{getLinkInfo}(c, j)$ ;
5:   if  $((l_{w,t} < \infty)$  and  $(l_{Bw,t} \geq B_{w,c,t}))$  then
6:      $W_j = l_{w,t}$ ;
7:   end if
8: end for
9: while  $S$  not empty do
10:  Node  $Min = \text{lower}(W)$ ;
11:   $l = \text{getLinkInfo}(c, Min)$ ;
12:  if  $(l_{Bw,t} < B_{w,c,t})$  then
13:    return Fail  $\triangleright$  Can not meet the requirements;
14:  end if
15:   $S' = S - \{Min\}$ ;
16:  for all Node  $j \in S'$  do
17:     $l = \text{getLinkInfo}(Min, j)$ ;
18:    if  $((l_{w,t} < \infty)$  and  $(W_j > W_{Min} + l_{w,t})$  and
19:     $(l_{Bw,t} > B_{w,c,t}))$  then
20:       $W_j = W_{Min} + l_{w,t}$ ;
21:    end if
22:  end for
23: end while
24: Slice  $f = \text{sliceBuilding}(W)$ 
25: return  $f$   $\triangleright$  Containing the configuration of the slice
```

Between lines 15 and 21 the algorithm iterates through the set of unprocessed nodes S' verifying if the current best cost node (Min obtained by the *lower* function) can be used to reach other nodes with lower cost. If Min does not have the requested Bw for the period ($B_{w,c,t}$), the algorithm checks that it can not meet the client requirements, then the search process is interrupted (line 13). When the most suitable node is identified, the algorithm travels through every node verifying if the Min node can reach the other nodes with a lower cost ($W_j > W_{Min} + l_{w,t}$).

In line 23 the output slice f is configured (function *sliceBuilding*) according to the definition of most suitable links in W (defined in the previous steps). Finally, in line 24 the slice f is returned as the result of the search algorithm. Algorithm 2 is $O(k \log k)$, since the search process occurs from one node to a subset of the non-analyzed remaining nodes.

C. Reliability Method

A network reliability method is necessary to calculate the suitability of the network slice defined by the algorithm

regarding the reliability requested in the SLA. It is worth to mention that the proposed R-REENC is agnostic of the reliability calculation method, i.e., different reliability calculation methods can be utilized with the proposed algorithm.

We decided to use the method proposed by Li et al. [17] as basis. The method assumes that, at any time, elements fail randomly and independently from each other, according to certain probabilities previously known, where a network with n components can have 2^n states. However, when the number of unreliable components is high, the computational cost of this method is exponential. This paper applied an alternative approach to enumerate the most probable states of the network (called m), providing a good approximation of the network reliability. We enhanced the Order algorithm so that the reliability can include only the communication paths between the client's node and all destination nodes. As result, the reliability method complexity is $O(n^2m + nm \log m)$, since it first computes the most probable states and then bounds the reliability of them.

Thus, the applied reliability method has the following advantages: (i) it is proposed for generic networks, i.e., it does not depend on information about physical devices or a database about the network infrastructure; (ii) it reaches a suitable approximation for the reliability with low computation time; (iii) it is flexible to vary the number of failures in the network; and (iv) it is easily adjustable to consider several nodes communication analysis.

Initially, the network slice structure is determined, and after that the reliability of the network slice can be calculated. This paper uses the reliability calculated as a criterion to constraints the suitability of the network slice. We consider the probability/capacity of the network links that interconnect all desired nodes, represented by a value in the range $[0, 1]$ (higher values result in higher reliability).

V. EXPERIMENTS

This section shows the evaluation of the proposed algorithm regarding the definition of network slices. In order to perform the experiments, a simulator¹ was implemented to enable the allocation of the network slices, where the metrics used for the evaluation are: successful allocations of network slices, number of adaptations, the saturation of network resources and the availability of bandwidth in the network infrastructure. Section V-A presents the scenario defined in the experiments and Section V-B discusses the results of the evaluation.

A. Scenario Configuration

During the evaluation process, several existing network topologies were used to measure the capacity of the proposed algorithm to define suitable network slices. These existing topologies have distinct structures, where the information about these topologies are available on the Internet². The following topologies were used: Internet2 (26 nodes and 31 links) and GEANT (39 nodes and 58 links). These topologies

are considered robust, being suitable options to be used in the performance evaluation [18], [19], [20].

In the experiments the network slices have the following parameters: (A) a set of nodes to be connected (randomly chosen); (B) duration of requests; and, (C) 24 bandwidth values assigned to the periods of the day (with average bandwidth request of 100Mbps). One hundred sets of requests were randomly generated, where each set was composed of client's requests, where each request has 24 network slices, representing the schedules of the day (i.e., $T = 24$).

The inter-arrival time and duration of network flows follow an exponential distribution and are the basis to model the bandwidth demand [21]. Therefore, in this paper, the requested bandwidth and the duration of the requests are generated by an exponential distribution, since both parameters can be considered related to the bandwidth demand. Additionally, all the results are presented with a 95% confidence interval.

B. Results

This section presents the results of the performed experiments, comparing the performance of the proposed R-REENC algorithm with the existing Per Time Calculation (PETIC) [8], since, according to the experiments in reference [8], PETIC already presented a better performance than the other existing approaches. In summary, PETIC individually evaluates the definition of network slices and does not apply a reliability strategy.

In this way, the experiments evaluated four approaches: (1) original PETIC; (2) PETIC with the proposed reliability strategy (Section IV), called R-PETIC; (3) the proposal of this paper without the reliability strategy, named REENC; and, (4) the complete proposal of this paper, the R-REENC algorithm. This organization aims to evaluate the benefits of each approach, seeking to meet the Bw demand per period of time, to minimize the adaptations and to reach the desired reliability of the network slice.

The results of the experiments are illustrated in the following figures, where three metrics are evaluated:

- Successful allocations (Figure 2): the number of requests solved, i.e., how many slices the strategy was able to allocate with the desired bandwidth.
- Saturated links (Figure 3): the number of links that have less than 10% of the original bandwidth available.
- Number of adaptations (Figure 4): the number of changes between slices of different periods, i.e., the number of network slices that modified its network elements from one period to another.
- Number of operational network slices post-failure (Figure 5): the number of network slices that keeps the connectivity in face of incremental failures in the network infrastructure.

Figure 2 presents the number of network slices that comply with the SLA (successfully allocated). Both R-REENC and R-PETIC have a similar performance when using or not the reliability strategy. Based on information from Figure 2, it is clear the advantage of the proposed reliability strategy, since

¹<https://bitbucket.org/rafaellgom/slicing-timeslot/>

²<http://www.topology-zoo.org/dataset.html>

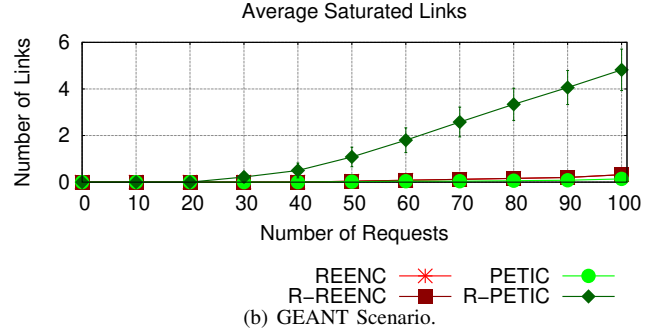
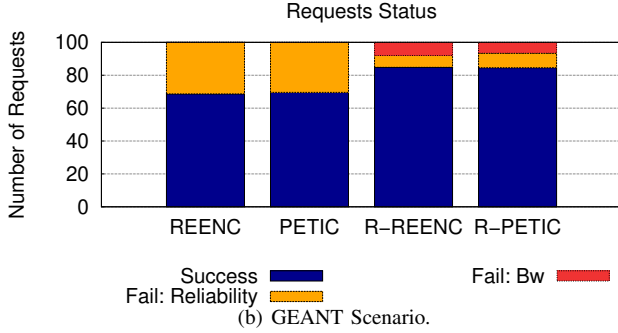
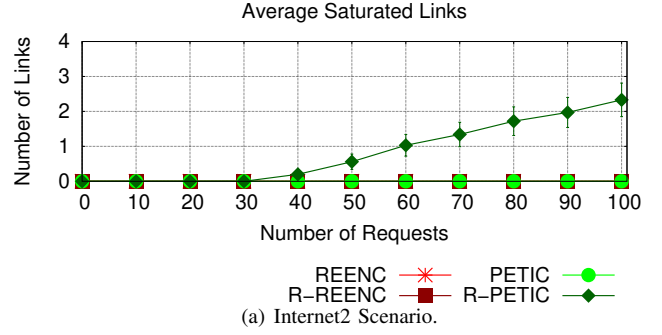
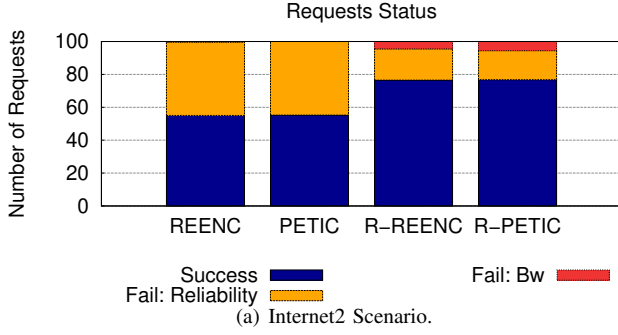


Fig. 2. Successful slice allocations.

Fig. 3. Saturated Links.

the algorithm without it has poorer performance, allocating around 30% less network slices. This improvement in the performance of the algorithms, when the reliability strategy is applied, occurs due to the capacity of the algorithms to meet the minimum reliability requested.

Additionally, it is important to mention that the reliability strategy, due to the path redundancy deployment, results in network slices with higher number of network elements. In this way, the application of the reliability strategy uses more resources of the network infrastructure, causing high Bw demand requests analyzed later not to be met (due to lack of resources). This fact does not happen when the algorithms are applied without the reliability strategy, whereas failures of network slices allocation keep the resource availability. This situation can be observed in Figure 3, which illustrates the number of saturated links.

According to Figure 3, except R-PETIC, all the algorithms have a small number of saturated links. This good performance is a consequence of the suitability weight metric applied (line 7 of Algorithm 1), which avoids the saturation of the Bw in the links. However, R-PETIC, even using the suitability metric, generates saturated links, because it does not consider the structure of previous network slices tending to use committed links in the redundancy. This fact does not happen in the R-REENC algorithm, resulting in a small saturation.

Regarding the number of adaptations generated by the algorithms, shown in Figure 4, the algorithms REENC and R-REENC presented the least number of adaptations performed, even when using the reliability strategy. On the other hand, R-PETIC has the greatest number of adaptations, in average four times the number of R-REENC. This behavior results from R-

PETIC's nature to define network slices individually, tending to generate network slices with distinct structures (even in unnecessary cases).

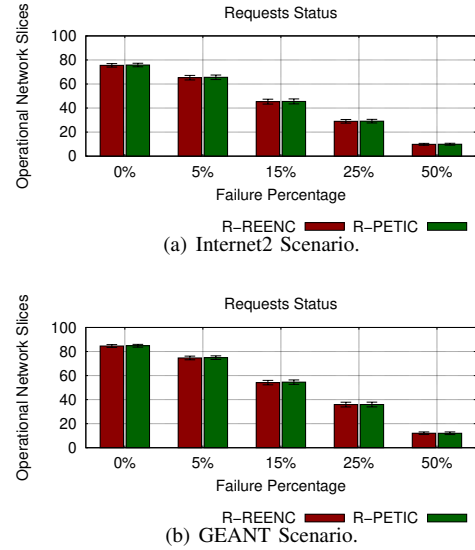


Fig. 5. Post-failure IAS status.

Figure 5 presents the number of network slices that keep connectivity after several failures in the network infrastructure. It represents the capacity of the analyzed algorithms to guarantee the performance of IAS delivery to the clients. Thus, only algorithms that apply the reliability strategy were evaluated, i.e., R-REENC and R-PETIC. The "x" axis in Figure 5 represents the percentage of network components that randomly fail. Thus, the point "0%" of Figure 5 illustrates

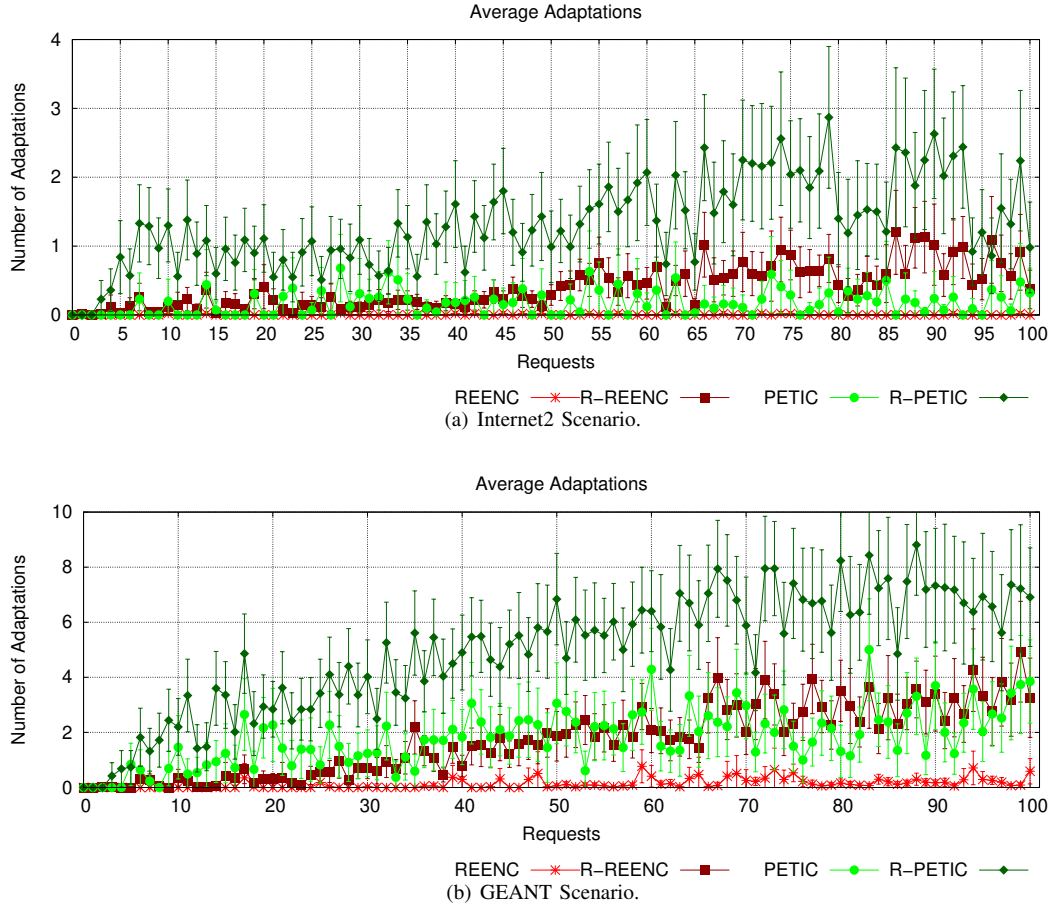


Fig. 4. Number of Adaptations.

the number of solved requests (e.g., number of successful allocations fulfilling the SLA) when the network infrastructure is fully operational. Similarly, the point “5%” represents the situation when 5% of the network elements fail, and so on.

Results from Figure 5 suggest R-REENC is suitable to define reliable network slices, since all slices provide a certain level of connectivity. The level of connectivity is directly related to the number of network elements taking part in a network slice, since the higher the number of network elements, the higher the cumulative probability of loss in the connectivity when a failure occurs. Additionally, it can be noted that the proposed reliability strategy maintains the network slices allocated, due to the survivability capacity of both algorithms when the reliability strategy is applied.

According to the results, the R-REENC algorithm evolves the network slice allocation process for ISPs. R-REENC overcame the existing solution from three perspectives: (I) a higher Bw availability in the network infrastructure; (II) the allocation of reliable network slices for the periods of time defined, and (III) the minimization of the number of adaptations necessary during the transitions of periods of time.

VI. CONCLUSION

The application of new technologies, such as NV, NFV and SDN, enhances the flexibility and management capacity of

ISPs, allowing a better utilization of network infrastructure and, consequently, providing higher quality services to their clients through the definition of suitable network slices. However, the definition of which network elements should be part of each slice is an open issue. Thus, this paper proposes an algorithm for allocating network slices considering the elastic bandwidth behavior and the reliability of these slices.

The proposed algorithm, called R-REENC, focuses on the allocation of network slices according to the traffic demand of clients during the day, while attending the minimum reliability requested in the SLA. The R-REENC algorithm overcomes the existing approach for individual slice definition, fulfilling a higher number of client requests (around 30%) and generating a lower number of necessary adaptations. This performance enables the maximization of profits of ISPs, as well as it improves the quality of IAS delivery.

ACKNOWLEDGMENT

This work was partially supported by CNPq (grants n. 307118/2017-7 and 405642/2016-4). This work is part of the INCT of the Future Internet for Smart Cities (CNPq 465446/2014-0, Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001, and FAPESP 2014/50937-1 and 2015/24485-9).

REFERENCES

- [1] R. Ferrús, O. Sallent, J. Pérez-Romero, and R. Agustí, "Management of network slicing in 5g radio access networks: Functional framework and information models," *CoRR*, vol. abs/1803.01142, 2018.
- [2] R. D. Doverspike, K. K. Ramakrishnan, and C. Chase, *Guide to Reliable Internet Services and Applications*. London: Springer London, 2010, ch. Structural Overview of ISP Networks, pp. 19–93.
- [3] C. Liang and F. R. Yu, "Wireless virtualization for next generation mobile cellular networks," *IEEE Wireless Communications*, vol. 22, no. 1, pp. 61–69, February 2015.
- [4] A. X., Z. C., T. R., G. R., K. A., S. D., and H. A., "On end to end network slicing for 5g communication systems," *Transactions on Emerging Telecommunications Technologies*, vol. 28, no. 4, 2017.
- [5] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network slicing in 5g: Survey and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 94–100, May 2017.
- [6] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Muñoz, J. Lorca, and J. Folgueira, "Network slicing for 5g with SDN/NFV: concepts, architectures and challenges," *CoRR*, vol. abs/1703.04676, 2017. [Online]. Available: <http://arxiv.org/abs/1703.04676>
- [7] H.-W. Lee, E. Modiano, and K. Lee, "Diverse Routing in Networks With Probabilistic Failures," *IEEE/ACM Transactions on Networking*, vol. 18, no. 6, 2010.
- [8] R. L. Gomes, F. R. P. da Ponte, A. Urbano, L. F. Bittencourt, and E. Madeira, "Slicing of network resources in future internet service providers based on daytime," in *2018 IEEE Symposium on Computers and Communications (ISCC) (ISCC 2018)*, Natal, Brazil, Jun. 2018.
- [9] R. Alvizu, G. Maier, N. Kukreja, A. Pattavina, R. Morro, A. Capello, and C. Cavazzoni, "Comprehensive survey on t-sdn: Software-defined networking for transport networks," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2232–2283, Fourthquarter 2017.
- [10] R. Vilalta, A. Mayoral, R. Munoz, R. Casellas, and R. MartÑez, "Multitenant transport networks with sdn/nfv," *Journal of Lightwave Technology*, vol. 34, no. 6, pp. 1509–1515, March 2016.
- [11] J. Son and R. Buyya, "Latency-aware virtualized network function provisioning for distributed edge clouds," *Journal of Systems and Software*, vol. 152, pp. 24 – 31, 2019.
- [12] M. Leconte, G. S. Paschos, P. Mertikopoulos, and U. C. Kozat, "A resource allocation framework for network slicing," in *IEEE Conference on Computer Communications (INFOCOM)*, April 2018, pp. 2177–2185.
- [13] P. Caballero, A. Banchs, G. de Veciana, X. Costa-Pañez, and A. Azcorra, "Network slicing for guaranteed rate services: Admission control and resource allocation games," *IEEE Transactions on Wireless Communications*, vol. 17, no. 10, pp. 6419–6432, Oct 2018.
- [14] J. Kwak, J. Moon, H. W. Lee, and L. B. Le, "Dynamic network slicing and resource allocation for heterogeneous wireless services," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Oct 2017, pp. 1–5.
- [15] K. Han, S. Li, S. Tang, H. Huang, S. Zhao, G. Fu, and Z. Zhu, "Application-driven end-to-end slicing: When wireless network virtualization orchestrates with nfv-based mobile edge computing," 05 2018.
- [16] G. Wang, G. Feng, W. Tan, S. Qin, R. Wen, and S. Sun, "Resource allocation for network slices in 5g with network resource pricing," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, Dec 2017, pp. 1–6.
- [17] V. Li and J. Silvester, "Performance Analysis of Networks with Unreliable Components," *IEEE Transactions on Communications*, vol. 32, no. 10, pp. 1105–1110, oct 1984.
- [18] R. d. S. Couto, S. Secci, M. E. M. Campista, and L. H. M. K. Costa, "Reliability and survivability analysis of data center network topologies," *Journal of Network and Systems Management*, vol. 24, no. 2, pp. 346–392, Apr 2016. [Online]. Available: <http://dx.doi.org/10.1007/s10922-015-9354-8>
- [19] D. F. Rueda, E. Calle, and J. L. Marzo, "Robustness comparison of 15 real telecommunication networks: Structural and centrality measurements," *Journal of Network and Systems Management*, vol. 25, no. 2, pp. 269–289, 2017.
- [20] S. Knight, H. X. Nguyen, N. Falkner, R. Bowden, and M. Roughan, "The internet topology zoo," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 9, pp. 1765–1775, October 2011.
- [21] T. M. Chen, "Network Traffic Modeling," pp. 326–339, 2007.