Re-optimizing the embedding of virtual infrastructures

João Soares and Susana Sargento

***Abstract*— Today, in the cloud computing context, the network is starting to arise as a true resource itself and not just as a required connectivity add-on. The ability to define network resources (e.g. routing/switching elements, bandwidth) in this context is still scarce, but there are clear evidences that this is a future reality. In this article, we consider that cloud infrastructure services will allow the definition of complete infrastructures, to which we refer as Virtual Infrastructures (VIs). These VIs comprise computing, storage and network resources. This article specifically tackles the VI embedding problem, which is known to be NP-hard. Having in mind this fact, different embedding strategies based on Integer Linear Programming (ILP) formulation are presented. The problem is addressed by proposing strategies that target load balancing and energy consumption of physical resources along with re-optimization strategies. A thorough evaluation of the different strategies is performed, studying the VI acceptance ratio, physical infrastructure energy consumption, and the impact of reconfigurations. Results show that there is a clear tradeoff between improving the VI acceptance ratio and reducing the physical infrastructure usage without allowing re-optimizations. We study in-depth different ways of combined strategies.**

***Index Terms* - Cloud Networking, Integer Linear Programming, Virtualization, Virtual Infrastructure.**

I. INTRODUCTION

THE network has a fundamental role in the cloud computing context, since it is able to guarantee performance, reliability and security to cloud services both inside Data Centers (DCs) as well as outside, i.e. on the Wide Area Network (WAN). The momentum around the Network Functions Virtualization (NFV) [1] concept can be seen as the most recent catalyst to this uprising need for an active role of the network in the cloud. The Telco sector seems eager to "cloudify" some of its solutions, which have more than just computing requirements: they have very strict requirements in terms of network guarantees. In fact, some of these solutions are already "cloud-ready", e.g. Evolved Packet Core (EPC) and IP Multimedia Subsystem (IMS). However, today's cloud solutions cannot meet the requirements imposed by these systems, mainly in terms of network aspects.

These recent events have re-leveraged the concept of Network-as-a-Service (NaaS). The "as-a-Service" concept is, among other aspects, related with dynamics and agility, which is on the birth of the concept of Software Defined Networking (SDN). Among the opportunities that SDN brings is the effective "unlock" of the NaaS concept. In this scenario of revolution and evolution, the IaaS and NaaS concepts will naturally tend to merge, which will require an integrated management of resources across cloud and network providers.

This article addresses the joint embedding of cloud and network resources, a NP-hard problem, through ILP. We consider the provisioning of computing and network resources through virtualization, enabling the support of VIs. The VI embedding problem is addressed in a single domain perspective, where a pool of computing and network resources (i.e. server and network nodes) is managed in an integrated way by a single entity. An example of such entity can be a Cloud Provider DC or a network operator with computing resources spread throughout its network.

We propose different VI embedding strategies that consider load balancing of resources, energy consumption, and the impact of re-optimization processes. The different strategies are submitted to a thorough evaluation. It is shown that allowing reconfiguration improves the VI acceptance ratio as well as energy consumption, but that can impact running services. Therefore, our proposed strategy aims to also minimize the costs associated with the reconfiguration process. We thoroughly explore how enabling VI reconfigurations can benefit the VI embedding problem, without neglecting the impact of the reconfiguration processes. The following contributions are made:

- ILP formulation for the re-optimization of VIs and reconfiguration of resources;
- Formulation of the costs associated with re-optimization;
- Re-optimization embedding strategy for both load-balancing, green and dynamic load and green approaches;
- Evaluation of the re-optimization approaches with respect to their performance and associated costs.

The remainder of this article is organized as follows. Section II presents related work. Section III describes the scenario studied and elaborates on the embedding problem. Section IV proposes the mathematical formulations for the embedding, and section V proposes the different strategies. Further, section VI depicts and discusses the evaluation of the different strategies. Finally, section VII provides final conclusions and future work.

II. RELATED WORK

This section presents works that closely relate to the VI allocation problem.

In [5], we have first addressed the VI embedding problem. Different optimal strategies were proposed. In this work, we extend the formulation of [5] to enable VI reconfigurations in order to understand its impact in terms of acceptance ratio and energy consumption.

João Soares was with Portugal Telecom Inovação, Aveiro, Portugal, when performing this work, e-mail: (joaosoares@ua.pt)

Susana Sargento is with Instituto de Telecomunicações, University of Aveiro, Portugal, e-mail: (susana@ua.pt)

[6] presents a strategy that considers the ability to reconfigure currently mapped Virtual Networks (VNs) when trying to map a new one. The acceptance ratio of the VNs is evaluated, as well as the impact of the reconfigurations on the running services. VNs are characterized only by node CPU and link bandwidth. Nevertheless, this work was an important reference to the ILP formulations presented in this work with respect to re-optimization. [8] studies the delay minimization in the cloud network through a Mixed ILP (MILP) formulation. [9] addresses reconfiguration in order to maximize the energy savings, and propose two heuristic approaches benchmarked by MILP approaches. The trade-off between energy savings and delay minimization was later studied in [10]. However, just like in the previous works, the interplay between CPU, memory, storage and network resources is not considered. In [11] the authors designed a greedy algorithm with the objective to place applications where they will experience minimum network congestion. [12] has a similar approach but also takes into account the objective of maximizing the VMs fault/failure tolerance.

The joint manipulation of cloud and network virtual resources has not yet been widely explored. However there is a considerable amount of work in the VN embedding field, a problem that closely relates to the VI one, that is a good reference point for addressing the VI embedding problem.

III. CLOUD NETWORK VIRTUAL-ENABLED INFRASTRUCTURE

We consider the concept of cloud network virtual-enabled infrastructure, where cloud and network resources are part of a pool of resources, can be virtualized, and are managed in an integrated way. In this work, we look at the scenario where cloud and network resources are hosted within a DC or spread through an operator network.

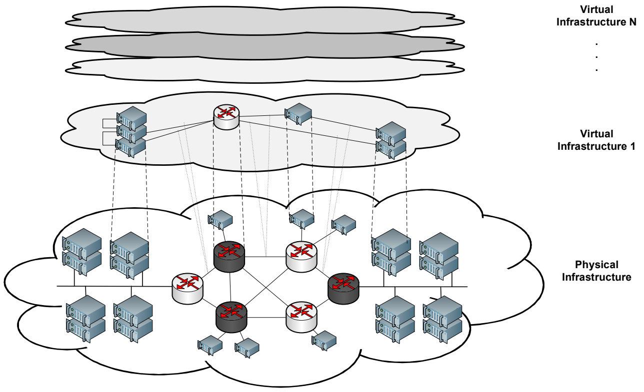

Fig. 1. Single Domain Physical Infrastructure view [5].

A. *Scenario*

Two types of nodes are considered, network nodes and server nodes, each with its specific set of associated characteristics. Network nodes are characterized by the number of CPUs, CPU clock frequency, and amount of memory (*Cs*, *F*, and *M*). Two types of network nodes are considered: those which have virtualization capabilities and can therefore host virtual network entities; and those that do not have virtualization capabilities, considered network transport elements only. Server nodes are characterized by the same parameters as network nodes plus storage capabilities (*STG*). With respect to links, these are characterized by bandwidth capacity (*B*) and assumed to be bidirectional and with a maximum delay (*D*).

The infrastructure can host multiple VIs, as it is illustrated in Fig.1. VIs are described similarly to physical infrastructures. VI nodes of a certain type (server, network) can only be accommodated in the infrastructure by nodes of the same type. Table I summarizes the notation. The reference to physical resources uses letter *P*, e.g., N^P, and virtual resources use letter *V*, e.g., $N^V{}_v$, where *v* refers to a specific VI. With respect to the connectivity of an infrastructure, an adjacent matrix is used: $A^P = N^P \rightarrow N^P$ when referring to a physical infrastructure. The convention used for the index notation is the following: *i*, *j* for nodes and links in the physical network, and *m*, *n* for nodes and links in the VI.

TABLE I. VI ASSIGNEMENT PROBLEM NOTATION – SINGLE DOMAIN

Symbol	Description
G^P	Physical Infrastructure
N^P	Set of Physical Nodes
S^P	Set of Server Nodes
R^P	Set of virtual-enabled Network Nodes
Rt^P	Set of transport Network Nodes
L^P	Set of Physical Links
i, j	Physical Nodes
ij	Physical Link
$Cs^P{}_i$	Number of CPUs of Physical Node *i*
$C^P{}_i$	Total CPU of Physical Node *i*
$M^P{}_i$	Total Memory of Physical Node *i*
$STG^P{}_i$	Total Storage of Physical Node *i*
$F^P{}_i$	CPU Frequency of Physical Node *i*
$B^P{}_{ij}$	Total Bandwidth of Physical Link *ij*
$D^P{}_{ij}$	Delay of Physical Link *ij*
$C^{Ptotal}{}_i$	Total CPU of Physical Node *i*
$M^{Ptotal}{}_i$	Total Memory of Physical Node *i*
$STG^{Ptotal}{}_i$	Total Storage of Physical Node *i*
$C^{Pfree}{}_i$	Free CPU of Physical Node *i*
$M^{Pfree}{}_i$	Free Memory of Physical Node *i*
$STG^{Pfree}{}_i$	Free Storage of Physical Node *i*

B. *Virtual Embedding Problem*

The VI embedding problem is known to be NP-hard. VIs can be mapped according to multiple objectives, e.g.: occupy the lowest bandwidth possible; balancing the occupation of the physical nodes. Combining both objectives is not simple as they are not standalone, i.e. the choice of an ideal physical node may not allow the choice of the ideal physical link. This is considered to be a multiple-objective optimization problem (MOP), where two or more conflicting objectives subject to constraints need to be simultaneously optimized.

IV. MATHEMATICAL FORMULATION

This section presents the mathematical formulation to solve the VI embedding problem. Special attention is given to the formulation introduced that allows the VI re-optimization.

A. *Assignment Variables & Generic Constraint*

The formulation considers two binary assignment variables, *x* and *y*, one for the virtual nodes and another for virtual links as equations (1) and (2) show.

$$x_i^{v,m}=\begin{cases}1\text{, virtual node } m \text{ of VI } v \text{ is allocated at physical node } i\\ 0\text{, else}\end{cases} \tag{1}$$

$$y_{ij}^{v,mn}=\begin{cases}1\text{, virtual link } mn \text{ of VI } v \text{ is uses physical link } ij\\ 0\text{, else}\end{cases} \tag{2}$$

As mentioned in section III, virtual servers are assigned to physical server nodes. Equation (3) reflects this constraint plus the fact that a virtual node is assigned just to one physical node. Moreover, equation (4) assures that virtual network nodes are assigned to network nodes with virtual capabilities.

$$\forall v,m\in S^P:\sum_{i\in S^P} x_i^{v,m}=1 \tag{3}$$

$$\forall v,m\in R^P:\sum_{i\in R^P} x_i^{v,m}=1 \tag{4}$$

Equations (5), (6) and (7) guarantee that the capacity values of physical nodes are kept within range. Equation (8) makes sure the physical node selected has at least the same frequency as the virtual node.

$$\forall i:\sum_{m,v} x_i^{v,m}\times C_{v,m}^V\le C_i^{P_{total}} \tag{5}$$

$$\forall i:\sum_{m,v} x_i^{v,m}\times M_{v,m}^V\le M_i^{P_{total}} \tag{6}$$

$$\forall i:\sum_{m,v} x_i^{v,m}\times STG_{v,m}^V\le STG_i^{P_{total}} \tag{7}$$

$$\forall i,v,m: x_i^{v,m}\times F_{v,m}^V\le F_i^{P_{total}} \tag{8}$$

Equation (9) applies the multi-commodity flow constraint [13] with a node-link formulation [14] to simultaneously optimize the mapping of virtual links and nodes. Moreover, the notion of direct flows on the virtual links is also used.

$$\forall v,\forall m,n\in N_v^V(m),m<n,\forall i:\sum_{j\in N^P(i)}\left(y_{ij}^{v,mn}-y_{ji}^{v,mn}\right)=x_i^{v,m}-x_i^{v,n} \tag{9}$$

Equation (10) guarantees that each physical link selected has enough bandwidth available to host a virtual link. Finally, equation (11) guarantees that the virtual link delay constraint is met.

$$\forall i,j\in N^P(i),i<j:\sum_{v,m,n\in N^V(m),m<n} B_{v,mn}^V\times\left(y_{ij}^{v,mn}+y_{ji}^{v,mn}\right)\le B_{ij}^{P_{free}} \tag{10}$$

$$\forall v,\forall m,n\in N_v^V(m),m<n:\sum_{\forall i,j\in N^P(i),i<j} D_{ij}^P\times\left(y_{ij}^{v,mn}+y_{ji}^{v,mn}\right)\le D_{v,mn}^V \tag{11}$$

B. Formulations

Herein, we specifically focus on the re-optimization formulation. The remaining formulations that support this article - maximum node and link load formulation, bandwidth consumption, and node and link state formulation - are presented in [5] and summarized here. The maximum load formulation provides the load consumption of each type of resource (server node, network node and link) that is more loaded among all other resources. The bandwidth consumption formulation calculates the total bandwidth that a VI consumes. The node and link state formulation keeps track of the state (active or not) of all resources.

The ability to completely reconfigure (re-arrange) currently mapped VIs when trying to map a new VI increases the likelihood of a successful mapping. In this sense, we keep record of the currently mapped VIs in the physical infrastructure in variables X and Y. $X_i^{v,m}$ denotes if virtual node m of VI v is mapped at the physical node i ($X_i^{v,m}=1$) or not ($X_i^{v,m}=0$). $Y_{ij}^{v,mn}$ denotes if virtual link mn of the VI v uses the physical link ij ($Y_{ij}^{v,mn}=1$) or not ($Y_{ij}^{v,mn}=0$).

Completely reconfiguring VIs gives an upper bound for the VI acceptance ratio; however, this is done at a high cost. The disruption caused is extremely high, especially when virtual nodes are migrated. Equation (12) guarantees that VI nodes already embedded must remain in the same physical hosts. Equation (13) guarantees that virtual links already embedded remain intact. When equations (12) and (13) are not considered, reconfiguration of nodes and/or links is possible.

$$\forall v\in V^{old},\forall m\in N^V,\forall i\in N^P: x_i^{v,m}=X_i^{v,m} \tag{12}$$

$$\forall v\in V^{old},\forall m,n\in N^V,\forall i\in N^P,m<n: y_{ij}^{v,mn}+y_{ji}^{v,mn}=Y_{ij}^{v,mn} \tag{13}$$

We keep track of node and link changes within each VI. $X_{v,m}^{change}$ registers if virtual node m in VI v is moved ($X_{v,m}^{change}=1$) or not ($X_{v,m}^{change}=0$). This constraint is reflected in equation (14). $Y_{v,mn}^{change}$ registers if virtual link mn in VI v was reconfigured ($Y_{v,mn}^{change}=1$) or not ($Y_{v,mn}^{change}=0$), which is guaranteed by equations (15), (16) and (17). We assume that the number of changes in the physical links used by a virtual link is counted only once. Variables $Y_{v,mn}^{changeAUX}$ and K are auxiliary variables to help keep track of virtual link changes. $Y_{v,mn}^{changeAUX}$, in equation (15), provides the number of physical links affected by changing the embedding of virtual link mn. Equation (16) guarantees that, if $Y_{v,mn}^{changeAUX}=0$, then $Y_{v,mn}^{change}=0$. In equation (17) variable K is a sufficient large number to ensure that $Y_{v,mn}^{change}=1$ if $Y_{v,mn}^{changeAUX}\ge 1$.

$$\forall v\in V^{old},\forall m\in N_v^V:\sum_{i\in N^P}\left(X_i^{v,m}-x_i^{v,m}\right)\times X_i^{v,m}=X_{v,m}^{change} \tag{14}$$

$$\forall v\in V^{old},\forall m,n\in N_v^V,m<n:\sum_{i,j\in N^P}\left(Y_{ij}^{v,mn}-(y_{ij}^{v,mn}+y_{ji}^{v,mn})\right)\times Y_{ij}^{v,mn}=Y_{v,mn}^{changeAUX} \tag{15}$$

$$\forall v\in V^{old},\forall m,n\in N_v^V,m<n: Y_{v,mn}^{change}\le Y_{v,mn}^{changeAUX} \tag{16}$$

$$\forall v\in V^{old},\forall m,n\in N_v^V,m<n:\frac{Y_{v,mn}^{changeAUX}}{K}\le Y_{v,mn}^{change} \tag{17}$$

Furthermore, we define V_v^{Change} in equation (20) to count the number of affected VIs. Similarly to what was previously explained, equations (18), (19) and (20) reflect the associated constrains. V_v^{Change} assumes value 1 if VI v was subject of a re-optimization process, and 0 otherwise. $V_v^{ChangeAUX}$, in equation (18), provides the number of virtual nodes and links in VI v affected by changing (either by no longer being embedded in a certain physical resource, or by embedding in a new one) the embedding of these resources. Equation (19) guarantees that if $V_v^{ChangeAUX}=0$, then $V_v^{Change}=0$.

$$\forall v\in V^{old}:\left(\sum_{m\in N_v^V} X_{v,m}^{change}\right)+\left(\sum_{m,n\in N_v^V:m<n} Y_{v,mn}^{change}\right)=V_v^{ChangeAUX} \tag{18}$$

$$\forall v\in V^{old}: V_v^{Change}\le V_v^{ChangeAUX} \tag{19}$$

$$\forall v \in V^{old} : \frac{V_v^{ChangeAUX}}{K} \leq V_v^{Change} \tag{20}$$

Since moving nodes and links has a significantly different impact, we define two other counters for affected Vis. One based only on virtual node changes – equations (21), (22) and (23) – and another based only on virtual link changes – equations (24), (25) and (26). $V_v^{NodeChange}$ assumes value 1 if VI v was affected by a virtual node change (0 otherwise). The same is true for $V_v^{LinkChange}$ with respect to virtual link changes. Variables $V_v^{NodeChangeAUX}$, $V_v^{LinkChangeAUX}$, and K are used for the same purpose as in the cases before - to keep the values of $V_v^{NodeChange}$ and $V_v^{LinkChange}$ within boundaries (0 or 1).

$$\forall v \in V^{old} : \sum_{m \in N_v^V} X_{v,m}^{change} = V_v^{NodeChange\ AUX} \tag{21}$$

$$\forall v \in V^{old} : V_v^{NodeChange} \leq V_v^{NodeChange\ AUX} \tag{22}$$

$$\forall v \in V^{old} : \frac{V_v^{NodeChange\ AUX}}{K} \leq V_v^{NodeChange} \tag{23}$$

$$\forall v \in V^{old} : \sum_{m,n \in N_v^V : m<n} Y_{v,mn}^{change} = V_v^{LinkChange\ AUX} \tag{24}$$

$$\forall v \in V^{old} : V_v^{LinkChange} \leq V_v^{LinkChange\ AUX} \tag{25}$$

$$\forall v \in V^{old} : \frac{V_v^{LinkChange\ AUX}}{K} \leq V_v^{LinkChange} \tag{26}$$

V. EMBEDDING STRATEGIES

The mathematical formulations presented in the previous section are the foundation for the definitions of different embedding strategies. In this section, we present a set of embedding strategies divide in three main categories: base strategies, dynamic strategies, and reconfiguration strategies.

A. Base Strategies

These strategies are considered base ones because they will be part of the strategies presented in the following categories.

1) Load Balancing + Shortest Path - VIE-Opt-LB+SP

This strategy aims to minimize the maximum load of node and link resources by also considering the total bandwidth consumption of a VI. S_{load}^{max} and R_{load}^{max} refer to the maximum load consumption of server and virtual-enabled network nodes, respectively. L_{load}^{max} refers to the maximum load consumption of links, and B_{cons} refers to the total amount of bandwidth consumed by a VI. In this case, the weights w_1 and w_2 have equal values so that there is an equal balance between maximum node load and overall link load (by considering maximum link load and total bandwidth consumed).

$$min \quad \omega_1\left(S_{load}^{max} + R_{load}^{max}\right) + \omega_2\left(L_{load}^{max} + B_{cons}\right) \tag{27}$$

2) Green strategy - VIE-Opt –Green

This strategy aims to minimize the energy consumption of the physical infrastructure. The main objective is to minimize the number of active nodes (x_i^{Active}) as they are the main energy consumption source. The second objective is to minimize the number of active links (y_{ij}^{Active}) to reduce the overall energy consumption of the active nodes - equation (28). In this case, w_1 is sufficiently higher than w_2 so that the primary objective is to minimize the number of active nodes, and ε is a very small value so that another objective can be considered as a tiebreak objective, presented in the equation as *Strat*. The objective considered in *Strat* in the evaluation section is the load balancing with shortest path (*VIE-Opt-LB+SP* – equation (27)). Load balancing the active elements of the infrastructure will ease the future embedding of VIs without having to increase the number of active elements.

$$min \quad \omega_1\left(\sum_{i \in N^P} x_i^{Active}\right) + \omega_2\left(\sum_{i,j \in N^P, i<j} y_{ij}^{Active}\right) + \varepsilon(Strat) \tag{28}$$

B. Dynamic Strategies - Adjusting Weights

This strategy combines different objectives dynamically. It uses the green strategy (*VIE-Opt –Green*) and the load balancing with shortest path (*VIE-Opt-LB+SP*). In equation weights (z_1 and z_2) are not fixed along time, but change depending on the infrastructure state at the time of each VI arrival. We consider that if the infrastructure is loaded over a certain limit, the load balancing strategy is applied. If the load is below or equal to the limit, the green strategy is applied.

$$min \quad z_1(Strat\_1) + z_2(Strat\_2) \tag{29}$$

C. Re-optimization Strategies

The strategies defined in this category allow the reconfiguration of already deployed VIs. We present two types of strategies: those that allow only link re-optimization; and those that allow both link and node re-optimization. Taking advantage of the strategies presented in the previous subsection, the following strategies are defined:

- Load Balancing (Node and Link) + Shortest Path strategy with Link Re-optimization - VIE-ReOpt-LB+SP
- Load Balancing (Node and Link) + Shortest Path strategy with Node and Link Re-optimization - VIE-LinkReOpt-LB+SP
- Green strategy with Link Re-optimization - VIE-LinkReOpt-LB-R
- Green strategy with Node and Link Re-optimization- VIE-ReOpt –Green
- Green + Load Balancing with Link Re-optimization - VIE-LinkReOpt-D-Green-LB
- Green + Load Balancing with Node and Link Re-optimization - VIE-ReOpt-D-Green-LB

Node reconfigurations are the most expensive ones, while link reconfigurations can be easily achieved and, to a certain extent, be neglected when considering the use of technologies like SDN. With this in mind, we do not penalize link reconfigurations. The strategies that allow only the re-optimization of links keep the original objective functions. However, those that allow also node re-optimization suffer a change, presented in equation (30). Due to the cost of node reconfigurations, the objective functions first try to minimize the number of virtual node reconfigurations. ε is a very small value so that *Strat* is considered as the last objective.

$$min \quad \left(\sum_{v \in V} \sum_{m \in N_v^V} X_{v,m}^{change}\right) + \varepsilon(Strat) \tag{30}$$

Depending on the strategy used, *Strat* will assume different forms. For example, in the strategy VIE-ReOpt-LB+SP, *Strat* will assume the form of equation (29), which leads to the objective function presented in equation (31). In this case, since both virtual node and link reconfigurations are allowed, equations (12) and (13) are not considered. In the case of strategy VIE-LinkReOpt-LB+SP, equation (13) is considered to prevent virtual node reconfigurations.

$$min \left(\sum_{v \in V} \sum_{m \in N_v^V} X_{v,m}^{change} \right) + \varepsilon \left(\omega_1 \left(S_{load}^{\max} + R_{load}^{\max} \right) + \omega_2 \left(L_{load}^{\max} + B_{cons} \right) \right) \qquad (31)$$

VI. EVALUATION

Herein we provide a thorough evaluation of the different optimal embedding strategies presented in section V. To evaluate the performance of the proposed approaches, a Matlab + CPLEX simulator is used. For each run, the program generates a random physical infrastructure of 20 nodes and a set of requests of VIs (with a number of nodes between 4 and 14). Both physical infrastructure and the generated VIs have 70% of the nodes as servers. These values were chosen because, in a cloud computing context, the required computing node capacity is much higher than the network one. The reason for having a small physical infrastructure lays on the fact that it allows to reach a saturation point faster, which highlights better the impact of the different strategies. Details on the nodes and link parameters can be seen in Table II. Moreover, the VI request rates (λ) vary between 2 and 5 VIs per time unit (Poisson arrivals), and the average duration of the VIs is 20 time units (exponentially distributed duration).

The weight values used for the optimal strategies are the following: w_1 and w_2 are 0.5; ε is 0.001; z_1 and z_2 have dynamic values that change depending on the state of the infrastructure. If the maximum load of any resource (server/network node or link) is below 75%, *Opt-Green* is applied; otherwise, *Opt-LB+SP* is applied. 75% was chosen empirically due to good results in a large set of experiments.

Table II: Physical and virtual infrastructure parameters

		Physical Infrastructure	**Virtual Infrastructures**
Net Nodes	**Cs**	{2; 4; 6; 8}	{1; 2; 3; 4 }
	F(Hz)	{2.0-3.2 / 0.2 steps}	{2.0-3.2 / 0.1 steps}
	Memory	{2; 4; 6}(GB)	{64; 128; 256; 512}(MB)
Server Nodes	**Cs**	{8; 16; 32; 64}	{1; 2; 4; 8; 16; 32; 64}
	F(Hz)	{2.8-3.2 / 0.2 steps}	{2.8-3.2 / 0.1 steps}
	STG (GB)	{6400; 12800; 25600}	{100; 200; 400; 800; 1600}
	M (GB)	{256; 512; 1024}	{2; 4; 8; 16; 32; 64}
Links	**B (Mbps)**	{500-2000 / 500 steps}	{10-100 / 10 steps}
	D (ms)	{5-10 / 1 steps }	{0-40 / 5 steps}

We analyze the VI acceptance ratio as one of the main indicators for the performance of the algorithms. Moreover, the energy consumption of the physical infrastructure is also analyzed. Finally, the impact of the re-optimization processes in the VI is carefully evaluated. All values in the following graphics present a mean of 10 runs (with different substrate) with a confidence interval of 95%.

A. Virtual Infrastructure Acceptance Ratio analysis

Fig.2 compares the results of the strategies with and without link re-optimization. It is clear that the strategies allowing link reconfiguration do not present significant differences. This happens because these strategies are limited in terms of virtual link mapping alternative solutions that make a reconfiguration viable.

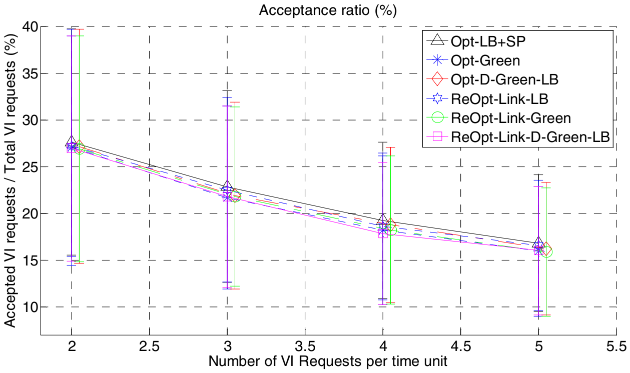

Fig. 2. VI acceptance ratio – with link re-optimization

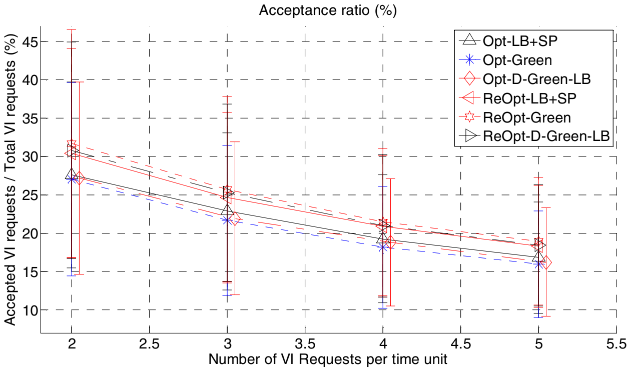

Fig. 3. VI acceptance ratio – with node and link re-optimization

On the other hand, the strategies allowing both node and link re-optimization outperform all others – see Fig.3. The values are, in average, up to 5% higher. The ReOpt-Green strategy is the one presenting the best performance (32%-19%), followed by the ReOpt-D-Green-LB and the ReOpt-Lb+SP. The difference between the three strategies is low (~1%). We notice that, in theory, all three strategies should present the exact same values in terms of acceptance ratio as they all allow full reconfiguration of resources. This does not happen due to the time constraint set to solve the ILP problem (600 seconds). In other words, in some cases no solution is found within the defined timeframe.

B. Physical Infrastructure Energy Consumption

Herein it is analyzed the performance of the different strategies in terms of the physical infrastructure energy consumption. While in terms of acceptance ratio some strategies present values relatively close, the differences with respect to energy consumption are more pronounced. The analysis is done considering the cumulative time of active resources, and the results are depicted in Fig. 4 and Fig. 5. The values are presented in percentage, where 100% means that all resources were active during the entire simulation time.

Fig.4 analyzes the values related to nodes as it is the lead indicator of the energy consumed. Not surprisingly, all strategies that consider the green policy outperform the remaining ones in terms of energy consumption, i.e. they present lower values. The VIE-ReOpt-Link-Green strategy clearly outperforms the remaining ones (47%-57%). The next strategy is the VIE-Opt-Green strategy (50%-60%). It is important to highlight that these two strategies perform very similarly in terms of acceptance ratio, but by allowing link reconfiguration, the VIE-ReOpt-Link-Green presents gains up to 3% with respect to active nodes. These strategies are then followed by VIE-ReOpt-Green (53%-63%) and the VIE-ReOpt-Link-D-Green-LB (53%-63%), followed by the VIE-Opt-D-Green-LB (55%-64%). Although having close values, it is important not to forget that the VIE-ReOpt-Green strategy presents gains up to 5% in terms of acceptance and still manages to present lower values in terms of active nodes. The VIE-ReOpt-D-Green-LB is the following strategy presenting better performance (61%-71%). The remaining optimal strategies, those that do not apply the green strategy, present higher and very close values (71%-81%).

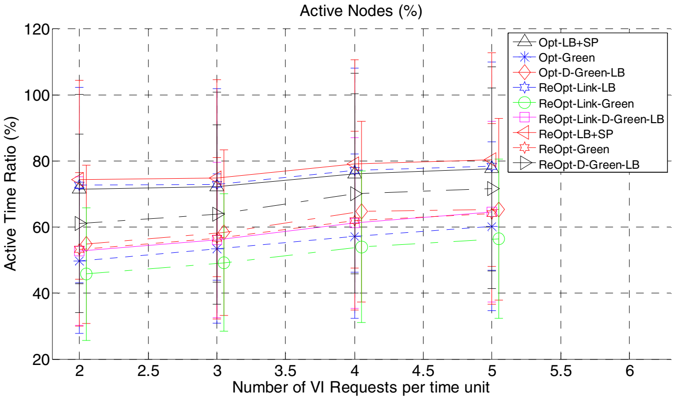

Fig. 4. Cumulative time of active nodes

We highlight here the fact that, among the strategies that consider energy consumption, those that allow only link re-optimization present similar acceptance ratio to those not allowing re-optimization. However, the former ones do improve the energy consumption indicators. If the impact of link reconfiguration is low to the point that it can be neglected, these strategies should indeed be considered as better approaches than those performing no reconfiguration.

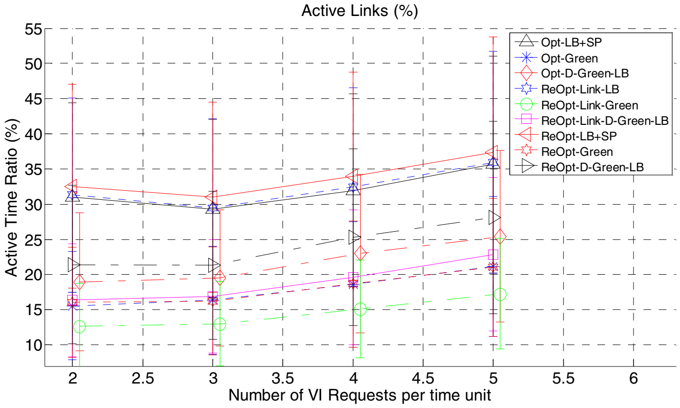

Fig. 5. Cumulative time of active links

In terms of active links (Fig.5) the strategies have similar performance. The optimal strategies with green approaches perform better, with values of active time between 12% and 27%. The optimal strategies with load-balancing policies present values between 29% and 37%. Note that, just like the acceptance ratio indicator, the values here present a linear behavior as the VI arrival rate increases. However, as opposite to the acceptance ratio, here they increase as the VI arrival rate increases. This is due to the fact that, although the acceptance ratio decreases with the increase of the VI arrival rate, the overall number of VIs embedded in the physical infrastructure increases, which leads to a higher percentage of active resources in the substrate.

C. *Re-optimization impact analysis*

In this subsection, it is performed an analysis to the different strategies that consider re-optimization of resources. The analysis is done in different perspectives: i) VIs affected by reconfiguration, whether by link or node; ii) VIs affected by link reconfiguration; iii) VIs affected by node reconfiguration; iv) overall number of link reconfigurations; v) overall number of migrated nodes.

Fig. 6 shows the number of VIs affected by reconfigurations (node and/or link). The VIE-ReOpt-D-Green-LB presents the highest value with a mean of 1.8 VIs affected per re-optimization process, followed by the VIE-ReOpt-LB+SP with a mean of 1.6. Then, it follows the strategies VIE-ReOpt-Green, VIE-ReOpt-Link-LB and VIE-ReOpt-Link-D-Green-LB with mean values of affected VIs of 1.3, 1.2 and 1, respectively. The strategy that shows lower impact is the VIE-ReOpt-Link-Green, with a mean of 0.6 affected VIs.

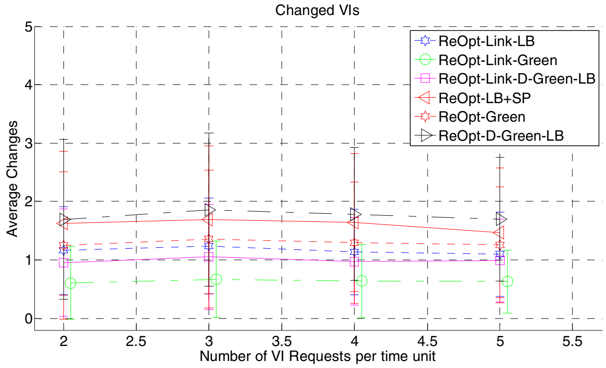

Fig. 6. VIs affected by changes

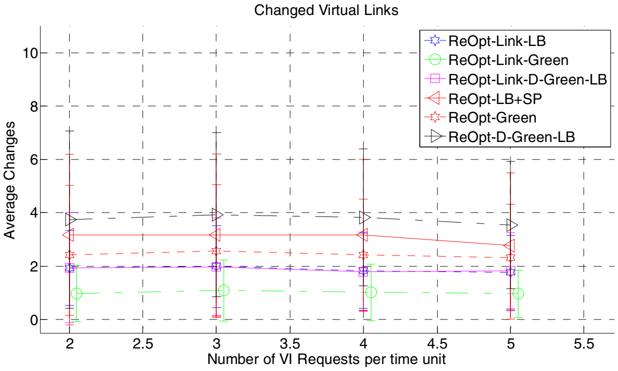

Fig. 7. Virtual link changes

The gap between the VIE-ReOpt-Link-Green and the other two strategies that allow only link reconfiguration is due to the fact that the former is a pure green strategy that tends to load active resources (nodes and links), which leads resources to a point of saturation much faster than the load balancing strategy. Therefore, with a higher probability of saturating nodes, trying to reconfigure virtual links to map a new VI will be in some cases inglorious. On the other hand, in the load balancing strategy, the probability of nodes reaching a saturation point is much lower than in the former case, which will allow more fruitful reconfigurations of virtual links.

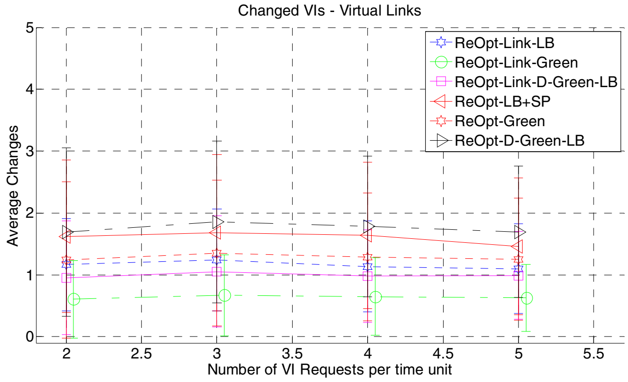

Fig. 8. VIs affected by virtual link changes

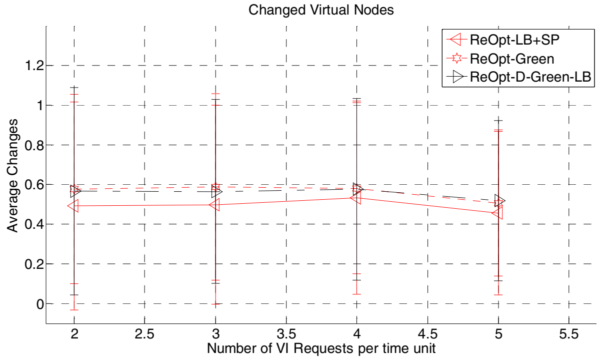

Fig. 9. Virtual node changes

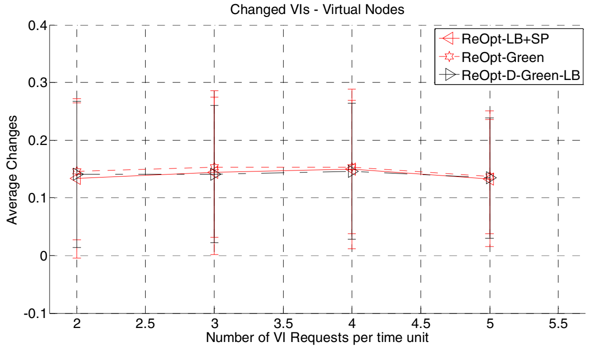

Fig. 10. VIs affected by virtual node changes

Fig. 7 shows the values related to the overall number of links reconfigured. The ranking here is the same as before. VIE-ReOpt-D-Green-LB (~3.9) is the strategy that triggers more link reconfigurations, followed by the VIE-ReOpt-LB-SP (~3.1), VIE-ReOpt-Green (~2.5), VIE-ReOpt-Link-LB (~1.9), VIE-ReOpt-Link-D-Green-LB (~1.9), and finally VIE-ReOpt-Link-Green (~1). The number of VIs affected by the reconfiguration of links is depicted in Fig. 8. Here the ranking remains the same as before. With respect to the number of nodes reconfigured, the values between the three strategies that allow node reconfiguration are close – see Fig. 9. The mean values are between 0.5 and 0.6 of virtual nodes that change. However, the number of VIs affected by the migration of nodes is only nearly 0.15 as shown in Fig. 10.

These results show that the node re-optimization impact can be low. However, it is considered as future work the characterization of VI nodes in terms of their ability/cost to me migrated. In this sense, the embedding strategy can better assess which nodes it should first try to migrate.

VII. CONCLUSION AND FUTURE WORK

This work focused on the VI embedding problem in a single domain. Optimal formulations and strategies based on ILP to solve the problem were proposed. An in-depth comparative performance analysis between different strategies was done. Strategies allowing full re-optimization of resources clearly outperform the remaining ones in terms of acceptance ratio. However, the adoption of these strategies needs to take into consideration the costs of reconfiguring resources. The reconfiguration impact values are optimistic. Furthermore, since each VI and resource can have specific requirements, an evolvement of these strategies could be to classify resources with respect to their ability/cost to be migrated. This aspect can be especially important in virtual nodes, since they are the ones more difficult to move.

We believe that the strategies presented in this work provide a solid base foundation that can be extended for more specific scenarios as the one just mentioned, and for scenarios considering other cloud properties like elasticity and scalability. In the cases of strategies allowing only link re-optimization, the values of acceptance ratio remain nearly the same; however, the energy consumption can be improved with green objectives. These are strategies worth considering, since the cost of reconfiguring links is much lower than of reconfiguring nodes.

REFERENCES

[1] ETSI, Network Functions Virtualisation (NFV): Architectural Framework, Technical Report ETSI GS NFV 002 v1.1.1, Oct. 2013.

[2] N. Yong; et.al, Connectivity as a Service: Outsourcing Enterprise Connectivity over Cloud Computing Environment, Computer and Management, International Conference on, 2011.

[3] W.H. Liao; S.C. Su, A Dynamic VPN Architecture for Private Cloud Computing, IEEE UCC, 2011.

[4] A. Mahimkar, et.al. Bandwidth on demand for inter-data center communication. ACM Workshop on Hot Topics in Networks, 2011.

[5] J. Soares, J. Aparicio, S. Sargento, Dynamic strategies for the optimal embedding of virtual infrastructures, submitted to IEEE ICC, June 2017.

[6] P.N. Tran, et.al, Optimal mapping of virtual networks considering reactive reconfiguration, 1st IEEE CloudNet, November 2012

[7] I. Houidi, et.al, Virtual network provisioning across multiple substrate networks, Computer Networks, Volume 55, Issue 4, 10 March 2011.

[8] B. Kantarci, H. Mouftah, Minimizing the provisioning delay in the cloud network: Benefits, overheads and challenges. IEEE ISCC, 1-4 July 2012.

[9] B. Kantarci, H. Mouftah, Optimal Reconfiguration of the Cloud Network for Maximum Energy Savings. IEEE/ACM CCGrid, 2012.

[10] B. Kantarci, H. Mouftah, Overcoming the energy versus delay trade-off in cloud network reconfiguration, IEEE ISCC, 1-4 July 2012.

[11] M. Mihailescu, et.al, Optimized application placement for network congestion and failure resiliency in cloudsm, IEEE CloudNet, 2015

[12] B. Spinnewyn, et.al, Fault-tolerant application placement in heterogeneous cloud environments, International CNSM, 2015.

[13] S. Even, et.al, On the complexity of time table and multi-commodity flow problems, Foundations of Computer Science, 16th Annual Symposium on, (pp. 184-193), 1975.

[14] M. Pioro, D. Medhi. Routing, Flow, and Capacity Design in Communication and Computer Networks. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc, 2004.